Author:
Dai, Xiaoyang

Title:
Bayesian methods for inferring selection and demographic from historical and contemporary DNA sequences

Bayesian methods for inferring selection and demographic from historical and contemporary DNA sequences

By

XIAOYANG DAI

Department of Biology
UNIVERSITY OF BRISTOL

A dissertation submitted to the University of Bristol in accordance with the requirements of the degree of DOCTOR OF PHILOSOPHY in the Faculty of Science.

DECEMBER 2019

Word count:

Abstract

In this thesis, I propose Bayesian methods to infer selection coefficients and allele age using time-series data and uncover the demographic history given contemporary whole-genome data.

Approximate Bayesian computation and Markov chain Monte Carlo method are widely used in solving population genetics problems. Time-series allele frequency problems often are modeled by the Hidden Markov Model, which is complex to make accurate inferences from. Here I employ a particle marginal Metropolis-Hastings method to make co-estimates of selection coefficients and allele age based on the single-locus Wright-Fisher model and the two-locus Wright-Fisher model. In addition, I also propose an EP method with the ABC algorithm to extract demographic information from whole-genome contemporary data.

For each method, I make simulation studies to present the accuracy of the method and apply the method to re-analysis of published data to show the method can achieve effective and accurate estimates for genetic parameters of interests.

Dedication and acknowledgements

I would like to thank my supervisor Mark Beaumont for his support and direction. His knowledge, time and tutoring has been crucial to my thesis, but most importantly his patience, positivity, kindness and calmness has been indispensable.

I thank my mother for filling my life with educational opportunities from an early age, and

I thank my father for teaching me that hard work pays off.

I thank Zhangyi He and Feng Yu for filling my days in Bristol with coffee, pizza, wine and friendship.

I thank Mengyao Zhang and Hui Yuan give me great support in my last period time of my writing-up.

I thank Qiang Wan, Can Liu, Xiaoyu Wang and and all my other friends and colleagues for making Bristol such a fabulous place to be.

Finally, I thank my grandparents for so much love they give to me.

Author's declaration

I declare that the work in this dissertation was carried out in accordance with the requirements of the University's Regulations and Code of Practice for Research Degree Programmes and that it has not been submitted for any other academic award. Except where indicated by specific reference in the text, the work is the candidate's own work. Work done in collaboration with, or with the assistance of, others, is indicated as such. Any views expressed in the dissertation are those of the author.

SIGNED: ..... DATE: .....

TABLE OF CONTENTS

List of Tables

List of Figures

Figure **Page**

CHAPTER 1

INTRODUCTION

The main focus of my PhD is to develop methods based on the Bayesian framework for detecting and characterizing the effects of selection in the genome. Efficient statistical inferences sometimes are very difficult to achieve in population genetics problems since large and complex data sets are involved. Even the simplest models will have many nuisance parameters which include genealogical information underlying the observations[15]. It is natural to consider using the Bayesian paradigm to solve these problems since it provides a flexible and diversity framework which can uncover the structure and parametrisation of a genetic model. However, Bayesian inference requires sufficient computational power to calculate out the likelihood function—the probability of obtaining the observations given some parameter value. In many circumstances, although it may be straightforward to build a computer program to simulate data, it may actually be very hard or impossible to figure out the likelihood function. This is often the case if the model has many hidden states and the probability of the data depends on summing probabilities over all possible states[12]. In the last decade, approximate Bayesian computation(ABC) has proved to be a highly flexible and likelihood-free Bayesian technique. It allows us to have an opportunity to make inferences under models of selection with or without considering the demographic structure. In this thesis, I introduce some ABC methods and Markov Chain Monte Carlo(MCMC) based methods which can be applied to solving demographic and selection problems based on time-series data and one-time point data.

1.1 Natural selection

1.1.1 Why understand natural selection is essential

Natural selection is one of the central mechanisms of evolutionary change and is the process responsible for the evolution of adaptive features. The basic idea of biological evolution is that populations and species of organisms change over time. Darwin suggested a mechanism for evolution: natural selection, in which heritable traits that help organisms survive and reproduce become more common in a population over time[24].

Without a working knowledge of natural selection, it is impossible to understand how or why living things have come to exhibit their diversity and complexity. An understanding of natural selection also is becoming increasingly relevant in practical contexts, including medicine, agriculture, and resource management. Natural selection is a central component of modern evolutionary theory, which in turn is the unifying theme of all biology. Without a grasp of this process and its consequences, it is simply impossible to understand, even in basic terms, how and why life has become so marvellously diverse[46].

1.1.2 What is natural selection

Natural selection is the differential survival and reproduction of individuals due to differences in phenotype. It is a key mechanism of evolution, the change in the heritable traits characteristic of a population over generations. Charles Darwin popularised the term "natural selection", contrasting it with artificial selection, which in his view is intentional, whereas natural selection is not[24].

Variation exists within all populations of organisms. This occurs partly because random mutations arise in the genome of an individual organism, and their offspring can inherit such mutations. Throughout the lives of the individuals, their genomes interact with their environments to cause variations in traits. The environment of a genome includes the molecular biology in the cell, other cells, other individuals, populations, species, as well as the environment. Because individuals with certain variants of the trait tend to survive and reproduce more than individuals with other less successful variants, the population evolves[24].

Natural selection acts on the phenotype, the characteristics of the organism which actually interact with the environment, but the genetic basis of any phenotype that gives that phenotype a reproductive advantage may become more common in a population. Over time, this process can result in populations that specialise for particular ecological niches and may eventually result in speciation. In other words, natural selection is a key process in the evolution of a population[24].

The effect of natural selection depends on genetic variation since it enables natural selection to increase or decrease the frequency of existing alleles in the population. There are many different sources of genetic variation such as mutation, random mating, recombination and so on. Genetic variation leads to an advantage of the interaction between organisms and environmental

changes since natural selection only acts on the phenotype and more genetic variation usually enables more phenotypic variation[62].

Another important mechanism in evolution is genetic drift which is caused by random sampling. In contrast to natural selection, the effects of genetic drift are not driven by environmental or adaptive pressures and lead to a random increase or decrease in allele frequency. Ronald Fisher held the view that genetic drift had a minor effect on evolution[39], where at the same time, Sewall Wright held that population structure and genetic drifts played an important role in evolution[127]. Motoo Kimura mentioned that the most important role is the combination of neutral mutations and genetic drift[62].

Researchers have taken different points of view on how selection works in genetics and the mechanism of selection in different species, however, the common view is that the understanding of selection is the core and important part to uncover the mechanism of evolution. Without knowledge of natural selection, it is impossible to understand how or why living things have come to exhibit their diversity and complexity. So understanding the level of natural selection is becoming more and more relevant in practical contexts[46]. Thanks to many sequencing technologies today, we have ample genomic variation data from plentiful individuals which gives us a good opportunity to have a deeper understanding of evolution.

1.2 Wright-Fisher model

1.2.1 Why we need Wright-Fisher model

The large amount and high quality of genomic data available today enable accurate inference of evolutionary histories of observed populations. The Wright-Fisher model is one of the most widely used models for this purpose. It describes the stochastic behaviour in time of allele frequencies and the influence of evolutionary pressures, such as mutation and selection.

The most basic and at the same time important model is the Wright–Fisher model for random genetic drift developed implicitly by Fisher (1922) and explicitly by Wright (1931). In its simplest version, it is concerned with the evolution of the relative frequencies of two alleles at a single diploid locus in a finite population of fixed size with non-overlapping generations under the sole force of random genetic drift, without any other influences like mutations or selection. The model can be generalised to multiple alleles, several loci, with mutations, selections, spatial population structures. For the basic two-allele case, this was first achieved in the important work of Kimura (1955), and he then went on to treat the case of several alleles (Kimura 1955, 1956). His solution, however, is local in the sense that it does not naturally incorporate the transitions resulting from the irreversible loss of one or several of the alleles initially present in the population. Consequently, the resulting probability distribution does not integrate to 1, and it is difficult to read off the quantitative properties of the process from his solution[113].

1.2.2 What is Wright-Fisher model

The Wright-Fisher model characterizes the evolution of a randomly mating population of finite size in discrete nonoverlapping generations. The model describes the stochastic behaviour in time of the number of copies (frequency) of alleles at a locus. The frequency is influenced by a series of factors, such as random genetic drift, mutations, migrations, selection, and changes in population size. When inferring the evolutionary history of a population, the effects of the different factors have to be untangled. Mutation, migration, and selection affect the allele frequency in a deterministic manner. We collectively refer to these as evolutionary pressures. The frequency also varies from one generation to the next as a result of random sampling of a finite-sized population random genetic drift. Mutations and migrations result in linear changes of the sampling probability, while selection is a non-linear pressure[63] [21] and therefore is more difficult to study analytically.

The Wright-Fisher model can be simulated in several ways. The most straightforward but not the most efficient approach is to keep track of individual genotypes in each generation and to randomly sample the parent of each individual from the previous generation. Alternatively, if we are only interested in the dynamics of the allele frequencies, then it suffices to just record these and to generate a binomially-distributed random variable $X_{t+1} \sim Binomial(2N, pt)$ and set $p_{t+1} = X_{t+1} \times \frac{1}{2N}$. However, when N is large, say $N \geq 1000$, even this approach may be too slow for many purposes. A crucial step for carrying out statistical inference in the Wright-Fisher model is determination of the distribution of allele frequency as a function of time, conditional on an initial frequency. Even though the Wright-Fisher model has a very simple mathematical formulation, no tractable analytical form exists for the distribution of allele frequency[85]. Therefore, various approximations have been developed, ranging from purely analytical to purely numerical. They generally either build on the diffusion limit of the Wright-Fisher model or rely on matching moments of the true distribution of allele frequency. Both types of approximations have been used successfully for inference of selection coefficients from time-serial data [74][115].

1.3 Monte Carlo method in population genetics

People can access genome data more and more easily and data are more abundant than ever. The advent of large-scale whole-genome variation data encourages people to tackle more complex population genetic models. It results in a challenge to statistical inference. Many inference problems in statistical genetics involve complex stochastic models that include a great number of variables. Besides that, some of the variables in the stochastic model, for example Wright-Fisher model, are not directly observable and are referred to as latent variables. The likelihood function for such inference problems can be expressed as the sum over the latent variables of the joint probability of the observed data and the latent variables, conditional on the genetic parameters of interest. Often, however, the space of latent variables is huge and that sum is not directly

computable [2].

To have a better understanding of evolution, we not only need to work with an enormous amount of data but we also need to compute the likelihood under more and more complex population genetic models. In particular, population genetic problems requires us to make inferences under increasingly high dimensional models and such models are often intractable, which means it is difficult or impossible to calculate their likelihood function. Because of that, standard methods are difficult to use and this circumstance provides great motivation to find some alternative powerful statistical approaches.

Monte Carlo methods are stochastic integration techniques that are useful for approximating such intractable sums. Approximation Bayesian computation and Markov chain Monte Carlo method are two important elements in the Monte Carlo family, which are widely used in population genetics problems to infer parameters of genetic interests. Here I will briefly introduce the advent of those two methods applied to population genetics and provide a review in more detail in chapter two.

1.3.1 Markov chian Monte Carlo in population genetic

The modern version of the Markov Chain Monte Carlo(MCMC) method was invented in the late 1940s by Stanislaw Ulam, while he was working on nuclear weapons projects at the Los Alamos National Laboratory. Immediately after Ulam's breakthrough, John von Neumann understood its importance and programmed the computer to carry out Monte Carlo calculations. Monte Carlo methods were central to the simulations required for the Manhattan Project, though severely limited by the computational tools at the time. In the 1950s they were used at Los Alamos for early work relating to the development of the hydrogen bomb, and became popularized in the fields of physics, physical chemistry, and operations research[80].

MCMC methods are primarily used for calculating numerical approximations of multi-dimensional integrals, for example in Bayesian statistics, computational physics, computational biology and computational linguistics. Since the 1980s, the use of MCMC methods has revolutionized the Bayesian analysis of complex statistical models, more details in book Robert and Casella [95]. Bayesian Markov c hainMonte Carlo and its many versions are algorithms to sample from a target distribution by using Markov chains, whose stationary or equilibrium distribution is an approximation of the target distribution. Its utility in population genetics was thrown wide open by a seminal paper by Kuhner, Yamato and Felsenstein[65]. There was followed up by several methods that have made use of MCMC algorithms for computing posterior density distributions of their parameters given genetic data, for example, Hey and Nielsen [53] used

Markov chain Monte Carlo simulations to integrate within the Felsenstein equation over the space of genealogies, whereas other parameters are integrated out analytically. Pritchard et al. [92] used Gibbs samplers to construct a model-based clustering method for using multi-locus genotype data to infer population structure. Beerli and Felsenstein [16] employed Markov chain Monte Carlo approach to investigate possible genealogies with branch lengths and with migration events. Since that time, the use of MCMC methods in statistical genetics has grown dramatically.

1.3.2 Approximation Bayesian computation in population genetics

Pritchard et al[91] initially introduced approximate Bayesian computation(ABC) to solve a population genetic problem in 1999. Since then, ABC has been well known as a method for intractable likelihood inference based on simulated data. It has been developed to avoid the requirement of likelihood functions and be widely used in biological sciences. ABC gained popularity last decade and has been applied to the analysis of many complex problems in different fields. Beaumont listed a number of good applications of ABC[13], including population genetics[105], ecology[57] , epidemiology[78], systems biology[69], anthropology[60], psychology[116], environmental modelling[22], climate modelling[54], and astronomy[48]. This wide range of application reflects that ABC has been widely accepted as a good option to work with a complex problem. And its flexible likelihood-free inference characteristics are in favour of many different fields.

1.4 Time series DNA data

As sequencing technologies progress, genome data sampled at multiple time become more accessible. Time series data arise from experimental evolution[20][122] or ancient DNA (aDNA) [71]. For example, Burke et al presented whole-genome resequencing data from sexually reproducing laboratory *Drosophila melanogaster* populations. These populations experienced over 600 generations of laboratory selection for accelerated development. However, newly arising advantageous alleles did not reach fixation after this number of generations. Burke et al thought that in wild populations the environment is unlikely to remain in strong natural selection for such long time. This suggests that selection should not be the only factor responding to expunge genetic variation in such sexual populations[20]. Besides experimental evolution data, there are many aDNA data used to study population genetics which is generated from archaeology. Allentoft et al use 101 Eurasia Bronze Age human samples to study human migrations and skin pigmentation phenotypic traits[1]. They show that the Bronze Age was a highly dynamic period involving large-scale population migrations and responsible for shaping major parts of present-day demographic structure in both Europe and Asia. Orlando et al use the fossil record of equids to study evolutionary processes of horse from early Middle Pleistocene to present[88]. The data they used is early to around 560–780 thousand years before present which is the oldest full genome sequence determined so far. Recently, Loog el at apply a Bayesian method to ancient

DNA of chickens [70]. They study time series data for loci TSHR and BCDO2 which show strong selection associated with a faster onset of egg laying and skin pigmentation respectively.

There are many other applications of using time series DNA data to investigate population genetics problem. Since under natural selection, the changes on genome data over time are regarded closely related to its strength, for example, aDNA can provide allele frequencies changes through time and it allows us to build a link between past selection with contemporaneous ecology directly. Based on that assumption, studying time series DNA data not only improves the performance of inference to selection coefficients but also helps with hypothesis tests comparing different models of selection fitness through time.

1.5 Aim of Thesis

This thesis intends to develop some Monte Carlo based methods which are employed to analyze time-series data focus on selection force and an ABC method which can be applied to whole-genome modern data focusing on demographic and population structure. In chapter two, I will demonstrate some basic concepts and intuitive ideas of MCMC and ABC. I also propose some ABC techniques and particle marginal Metropolis-Hastings method content, which is highly related to my later chapters. In chapter three, the method is mainly used to infer selection coefficient and allele age based on single-locus Wright-Fisher model. In chapter four, I develop another MCMC-based method to take recombination and linkage into account and to jointly infer selection coefficients for different loci based on linked-locus Wright-Fisher model. Based on expectation propagation method, I developed an ABC framework to make an inference on demographic and population structure which can analyze whole-genome modern data and present the results in chapter five.

CHAPTER 2

AN INTRODUCTION TO MONTE CARLO METHOD

Modern population genetics datasets require us to make inferences under increasingly high dimensional models and such models are often intractable, which means it is difficult or impossible to calculate their likelihood function. We need to employ Monte Carlo methods to make approximation of such intractable sums and approximation Bayesian computation(ABC) to realise intractable likelihood parameters inferences. In this chapter, I will briefly introduce what Markov chain Monte Carlo method and ABC are and present some examples with Markov Chain Monte Carlo(MCMC) and ABC techniques.

MCMC methods are used to calculate numerical approximations of high-dimensional integrals. This attractive feature makes the use of MCMC widespread, its applications in many fields, for example, computational physics, computational biology, Bayesian statistics and so on. Markov Chain Monte Carlo and its many versions are algorithms to sample from a target distribution by using Markov chains. In population genetics problems, MCMC is often used to compute the posterior distribution of parameters of interest from genetic data. For example, a very early application of the MCMC method in population genetics is from Ian J. Wilson and David J. Balding in 1998[125]. They tried to infer both historical events and evolutionary parameters by using stochastic simulations. They used MCMC to compute the likelihood by treating the ancestral allelic states as auxiliary parameters which made the computation process simplified. Later, MCMC was applied to estimate rates of recent immigration by given individual multilocus genotype data, where genotype frequencies are under the assumption of Hardy-Weinberg equilibrium proportions within populations[124].

ABC constitutes a class of computational methods rooted in Bayesian statistics that can be used to estimate the posterior distributions of model parameters. In model-based statistical inference, the likelihood function is of central importance, since it expresses the probability of the observed data under a particular statistical model, and thus quantifies the support data

lend to particular values of parameters and to choices among different models. However, for more complex models, an analytical formula might be elusive or the likelihood function might be computationally very costly to evaluate. In last twenty years, ABC has rapidly gained popularity in analysis of complex problems arising in biological sciences, e.g. in population genetics, ecology, epidemiology, and systems biology since ABC methods bypass the evaluation of the likelihood function.

In the last ten years, there were many ABC methods that have been implemented with MCMC such as ABC-MCMC[76], pseudo-marginal ABC[10], ABC-population Monte Carlo(ABC-PMC)[14] and so on. The variety of different MCMC based method introduced in ABC implies a close relationship between ABC and MCMC.

2.1 Markov Chain Monte Carlo method

2.1.1 Metropolis-Hastings algorithm

The basic intuition of the Metropolis-Hastings algorithm is published by Nicholas Metropolis et al in 1953[81]. It is used to generate a sequence of sample values and the more samples are generated, the approximation becomes closer to the target distribution $\pi(x)$. The sample is often drawn from a distribution which only depends on the current sample value. At each iteration, the Metropolis-Hastings algorithm draws a candidate sample value for the next iteration. With some probability(acceptance ratio), the candidate value is either accepted or not. If it is accepted, in the next iteration the candidate value is used as the current state; if it is not accepted, in next iteration the current state is still the current value in this iteration. We denote a symmetric proposal distribution with current state $\theta = \theta^{(t)}$ is $g(\cdot|\theta^{(t)})$, and $p(\theta|y)$ is the posterior distribution given by y. The Metropolis-Hastings algorithm proceeds as,

Algorithm 1 Metropolis–Hastings algorithm

Sample $\theta^{'} \sim g(\cdot|\theta^{(t)})$

Acceptance Ratio : $\alpha(\theta^{'}|\theta^{(t)}) = \frac{p(\theta^{'}|y)}{p(\theta^{(t)}|y)} = \frac{p(y|\theta^{'})p(\theta^{'})}{p(y|\theta^{(t)})p(\theta^{(t)})}$

$$\theta^{(t+1)} = \begin{cases} \theta^{'}, & \text{with probability } \min(\alpha,1) \\ \theta^{(t)}, & \text{otherwise} \end{cases}$$

The acceptance ratio α reflects how probable the new candidate sample is with respect to the current existing sample. If $\theta > 1$, we should include $\theta^{'}$ as it has a higher probability than current state $\theta^{(t)}$. If $\alpha < 1$, the ratio of $\theta^{'}$ compared to $\theta^{(t)}$ reflects the relative frequency of $\theta^{'}$ should be included. So we need to set $\theta^{(t+1)}$ is equal to $\theta^{'}$ with probability α and $\theta^{(t+1)}$ is equal to θ^{t} with probability $1-\alpha$. In the algorithm, this acceptance process is often accomplished by using an $u \sim$ Uniform(0,1) sampling and setting $\theta^{(t+1)} = \theta^{'}$ if $u < \alpha$, otherwise $\theta^{(t+1)} = \theta^{t}$ when $\alpha < 1$.

The sequence of samples generated from the Metropolis-Hastings algorithm is expected to be a Markov chain which asymptotically reaches a unique stationary distribution, i.e. the target distribution $\pi(\theta)$. If a Markov process has a unique stationary distribution $\pi(\theta)$, it is equivalent to satisfy both the existence of stationary distribution condition and uniqueness of stationary distribution condition. To be simplified, it requires distribution $\pi(\theta)$ to satisfy $\pi(\theta)q(\theta^{'}|\theta) = \pi(\theta)q(\theta|\theta^{'})$ and $\pi(\theta)$ is both aperiodic and positive recurrent at the same time. We denote target distribution $\pi(\cdot)$ and prior distribution $p(\cdot)$ here. Metropolis-Hastings algorithm simulates a Markov Chain with stationary distribution is the target distribution $\pi(\theta)$. We start the Metropolis-Hastings algorithm with an initial value sampled from the prior distribution. In the main loop, we generate new candidates based on a transition kernel $q(\cdot|\cdot)$ and we calculate the "criteria" whether to accept, the acceptance ratio α, at each iteration. We need to run the main loop until it converges.

Algorithm 2 MCMC-Metropolis–Hastings algorithm

Initialization: $\theta^{(0)} \sim p(\theta)$
for iteration $i = 1, 2, \cdots$ **do**
Candidate $\theta^{'} \sim q(\theta^{(i+1)}|\theta^{(i)})$
Acceptance Ratio : $\alpha(\theta^{'}|\theta^{(i)}) = \min\left\{1, \frac{p(\theta^{'})q(\theta^{(i)}|\theta^{'})p(y|\theta^{'})}{p(\theta^{(i)})q(\theta^{(')}|\theta^{(i)})p(y|\theta^{(i)})}\right\}$
$u \sim$ Uniform(0,1)
if $u < \alpha$
Accept proposal : $\theta^{(i+1)} = \theta^{'}$
else
Reject proposal : $\theta^{i+1} = \theta^{(i)}$
end if
end for

Here I use a toy example to illustrate how MCMC-Metropolis–Hastings algorithm works. If we want to have a chain from the target distribution $p(x)$, which is $p(x) = \beta \times N(-1,4) + (1-\beta) \times N(2, 0.25)$ where $N(\mu, \sigma^2)$ is Normal distribution with mean μ and standard deviation σ. β refers to how the target distribution will combine the two Gaussian distribution together. We regard its value varies from 0 to 1 in our examples. Firstly, we choose $\beta = 0.6$. The density of $p(x)$ for $\beta = 0.6$ is the plot (a) in Figure 2.1. I use a Normal distribution as a proposal and choose a far point as initial to start. The plot (b) in Figure 2.1 suggests the good performance of the Algorithm with limited prior knowledge. For 2000 steps, we can see the chain has started to converge. A histogram corresponding to this 2000 samples is shown in plot (c) in Figure 2.1 with true density curve, the most left bar represents my initial point. For 2000 samples, even without any burn-in or thinning methods, it starts to resemble the target distribution. For a longer run, I choose the number of steps is 50000, and the result is shown in plot (d) in Figure 2.1. The samples resemble the target distribution quite well.

To check how different value of β affect the performance of Metropolis–Hastings algorithm, especially the mixing, here I also present two more examples where $\beta = 0.1$ and $\beta = 0.9$ in

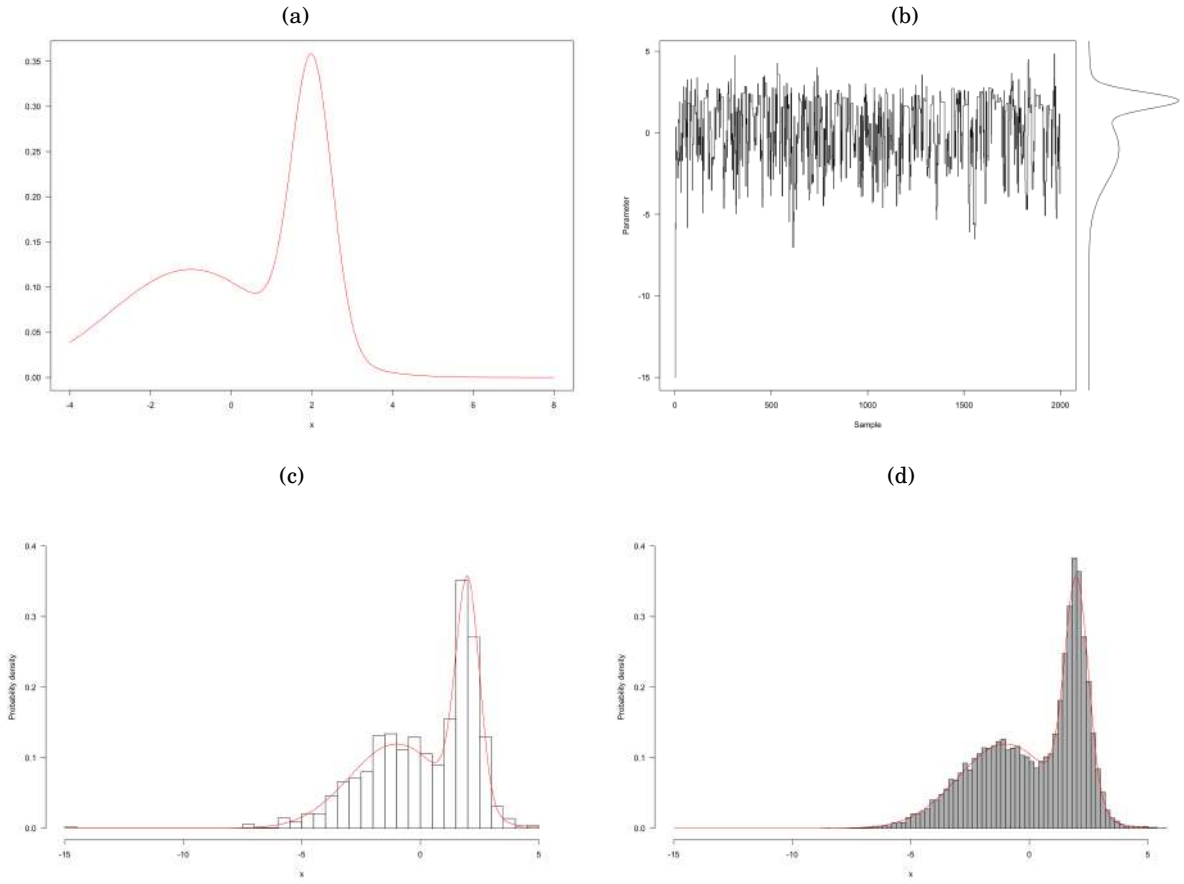

Figure 2.1: The toy example with MCMC-Metropolis–Hastings. (a) The density curve of target distribution $p(x) = 0.6 \times N(-1,4) + 0.4 \times N(2, 0.25)$. (b) The MCMC chain for 2000 steps with inital state $x = -15$ which is quiet far from the target and the right curve is the target density. (c) A 2000 samples and the red curve is the target density. (d) A 50000 samples and the red curve is the target density.

Figure 2.2. From Figure 2.2 (a) to (b), we can find the different value of β will affect the mixing significantly. It is reasonable that when $\beta = 0.1$, the target distribution will be more like $N(2, 0.25)$ which has less width of distribution. However, it still has a tail of distribution $N(-1,4)$ and it makes the Markov chains need to jump to a very unlikely state for distribution $N(2, 0.25)$. In inverse circumstance in Figure 2.2 (c) to (d), the Markov chain will much easier to be accepted due to the board distribution of $p(x) = 0.9 \times N(-1,4) + 0.1 \times N(2, 0.25)$ and the mixing is better than $\beta = 0.1$. For those examples, we can find that the performance of Metropolis–Hastings algorithm, especially the mixing performance of parameters, is highly related to the target distribution which suggests the tuning process is very essential to MCMC performance.

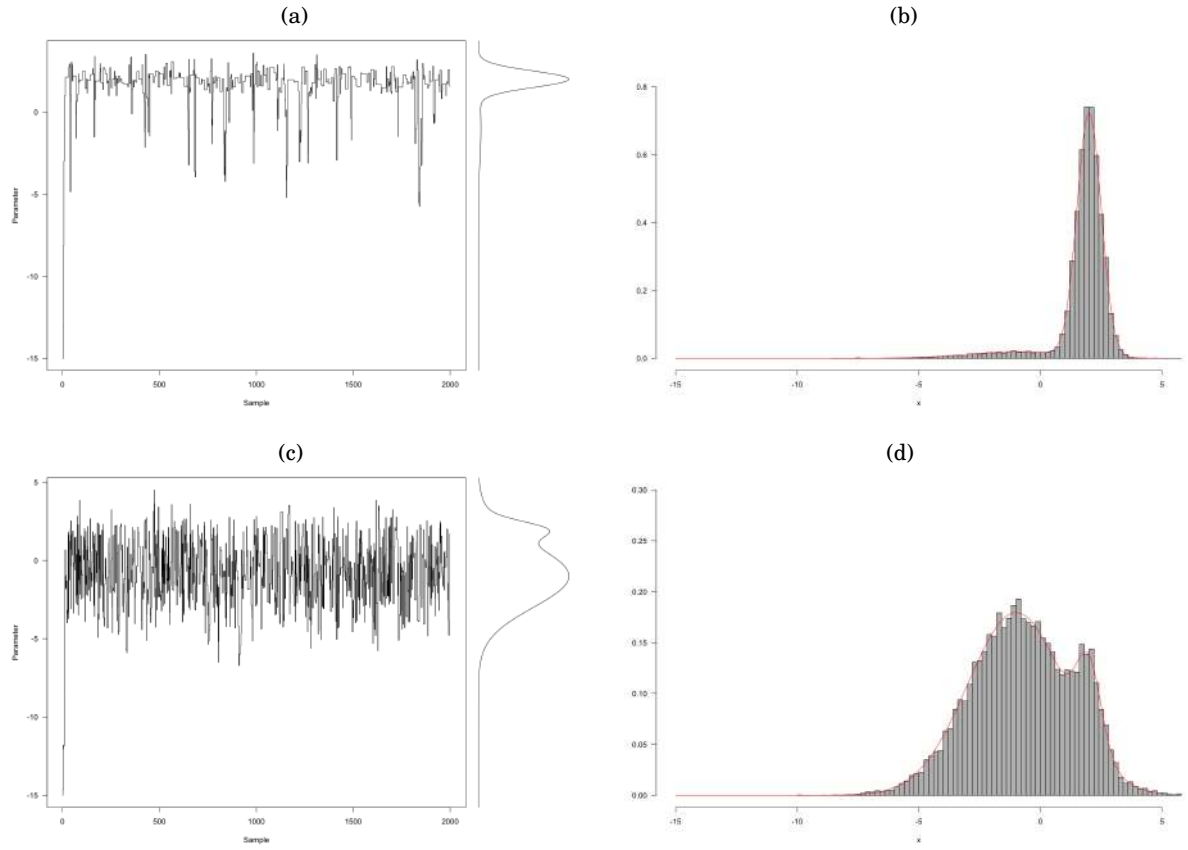

Figure 2.2: Two more example using $p(x) = \beta \times N(-1,4) + (1-\beta) \times N(2,0.25)$ MCMC-Metropolis–Hastings algorithm. (a) For $\beta = 0.1$, The MCMC chain for 50000 steps with initial state $x = -15$ and the right curve is the target density. (b) 50000 MC samples histogram with target density curve in red (c) For $\beta = 0.9$, The MCMC chain for 50000 steps with initial state $x = -15$ and the right curve is the target density. (d) 50000 MC samples histogram with target density curve in red.

2.1.2 Pseudo-Marginal Metropolis–Hastings algorithm

Pseudo-Marginal Metropolis–Hastings is an instance of the Metropolis–Hastings algorithm that extends its use to cases where the target density is not available analytically. It relies on the fact that the Metropolis–Hastings algorithm can still sample from the correct target distribution if the target density in the acceptance ratio is replaced by an estimate. It is especially popular in Bayesian statistics, where it is applied if the likelihood function is not tractable.

Supposing when using Metropolis–Hastings algorithm, it is often very hard to compute the key item in the acceptance ratio, the likelihood $p(y|\theta^{(i)})$. Especially, if there exist some unobservable processes or missing data, it is usually difficult to obtain a Monte-Carlo estimate for that likelihood efficiently. One typically way is to use MCMC schemes which target the joint posterior of the parameters and some auxiliary latent variables. However, these samplers can mix

poorly when latent variables and parameters are strongly correlated under the joint posterior distribution. Furthermore these schemes cannot be implemented if we can only simulate the latent variables and not evaluate their probability density function. Similarly, in the context of undirected graphical models, the likelihood function might involve an intractable integral over the observation space with examples from spatial statistics[82]. Pseudo-marginal MCMC is developed to solve this problem, it is only required to be able to make unbiased estimate of intractable likelihood distribution in the Metropolis–Hastings acceptance ratio and then using their unbiased estimates to continuously calculate the acceptance ratio. Pseudo-Marginal Metropolis–Hastings algorithm is quite often to be used in latent variable models, such as random effects models and state space models, where the likelihood can be estimated without bias using importance sampling[10] or particle filters[5].

Pseudo-Marginal Metropolis–Hastings algorithm was first introduced into statistics by Mark Beaumont in 2003. He used an importance sampling based approximate likelihood in MCMC to make estimations of recent changes in effective population size(N_e) using temporally spaced gene frequency data[10]. Later , Christophe Andrieu and Gareth Roberts studied on the properties of the Pseudo-marginal Metropolis-Hastings algorithm. They showed an unbiased non-negative estimate of likelihood, $\hat{p}(y|\theta)$; based on that estimate, the Pseudo-marginal MCMC algorithm will target to the true target distribution $\pi(\theta)$[5].

To be explicit, the acceptance ratio in MCMC is $\alpha(\theta^{'}|\theta^{(i-1)}) = \min\left\{1, \frac{p(\theta^{'})q(\theta^{(i-1)}|\theta^{'})p(y|\theta^{'})}{p(\theta^{(i-1)})q(\theta^{(')}|\theta^{(i-1)})p(y|\theta^{(t-1)})}\right\}$, and $p(y|\theta^{(i)})$ is not analytically available. If $\mathbb{E}[\hat{p}(y|\theta^{(i)})] = p(y|\theta^{(i)})$, then by using the estimate $\hat{p}(y|\theta^{(i)})$ will still result in the exact target posterior. For some cases, we often need to compute $p(y|\theta^{(i)})$ as an integral over some latent variables $x \in \mathbb{X}$. Assume the parameters we are interested in are $\theta \in \Theta$ and y is the partially observed data used to make inferences, then

$$\pi(\theta, x, y) = p(\theta)p(x|\theta)p(y|x,\theta) \tag{2.1}$$

$$p(y|\theta) = \int p(x|\theta)p(y|x,\theta)dx \tag{2.2}$$

We can make Monte Carlo estimates of the integral in 2.2 using $p(y|x,\theta)$ and random draws simulate from $p(x|\theta)$. That is, simulate a series of $x_1, x_2, \cdots, x_N$ with large N. So an exact approximating likelihood is given as the standard Monte Carlo estimate,

$$\hat{p}(y|\theta) = \frac{1}{N}\sum_{i=1}^{N} p(y|x_i,\theta) \tag{2.3}$$

We can also use an importance sampling scheme to draw a series of $x_1, x_2, \cdots, x_N$ from a proposal distribution $g(x|\theta)$, which is easier to access and has the same support with $p(x|\theta)$, then

$$p(y|\theta) = \int p(y|x,\theta)\frac{p(x|\theta)}{g(x|\theta)}g(x|\theta)dx \tag{2.4}$$

So we can sample a series of $x_1, x_2, \cdots, x_N$ from $g(x|\theta)$ and

$$\hat{p}(y|\theta) = \frac{1}{N}\sum_{i=1}^{N} p(y|x_i,\theta)\frac{p(x_i|\theta)}{g(x_i|\theta)} = \frac{1}{N}\sum_{i=1}^{N} p(y|x_i,\theta)\omega_i(p||g) \tag{2.5}$$

where $\omega_i(p||g)$ is known as importance weights. We can present a Pseudo-marginal Metropolis-Hastings algorithm as follows,

Algorithm 3 Pseudo-marginal Metropolis–Hastings algorithm

Initialization: $\theta^{(0)} \sim p(\theta)$
for iteration $i = 1, 2, \cdots$ **do**
Candidate $\theta^{'} \sim q(\theta^{(i)}|\theta^{(i-1)})$
Calculate approximate likelihood $\hat{p}(y|\theta^{(i-1)})$ and $\hat{p}(y|\theta^{'})$
Acceptance Ratio : $\alpha(\theta^{'}|\theta^{(i-1)}) = \min\left\{1, \frac{p(\theta^{'})q(\theta^{(i-1)}|\theta^{'})\hat{p}(y|\theta^{'})}{p(\theta^{(i-1)})q(\theta^{(\prime)}|\theta^{(i-1)})\hat{p}(y|\theta^{(t-1)})}\right\}$
$u \sim$ Uniform(0,1)
if $u < \alpha$
Accept proposal : $\theta^{(i)} = \theta^{'}$
else
Reject proposal : $\theta^{i} = \theta^{(i-1)}$
end if
end for

If we substitute $p(x|\theta)$ in the acceptance ratio by an importance sampling estimate, denote an independent samples $x_1, x_2, \cdots, x_n$ draw from the proposal $g(x|\theta)$, the acceptance ration in pseudo-marginal Metropolis-Hastings (MH) is

$$\alpha = (\theta^{'}|\theta^{(i-1)}) = min\left\{1, \frac{p(\theta^{'})q(\theta^{(i-1)}|\theta^{'})\hat{p}(y|\theta^{'})\prod_{i=1}^{n} g(x_i|\theta^{'})\prod_{i=1}^{n} g(x_i|\theta^{(i-1)})}{p(\theta^{(i-1)})q(\theta^{'}|\theta^{(i-1)})\hat{p}(y|\theta^{(t-1)})\prod_{i=1}^{n} g(x_i|\theta^{(i-1)})\prod_{i=1}^{n} g(x_i|\theta^{'})}\right\} \tag{2.6}$$

Then we can show the Metropolis–Hastings sampler as $p(\theta)\hat{p}(y|\theta)\prod_{i=1}^{n} g(x_i|\theta)$, which can exactly marginalise down to posterior $p(\theta|y)$ if given $\mathbb{E}[\hat{p}(y|\theta^{(i)})] = p(y|\theta^{(i)})$,

$$\int p(\theta)\hat{p}(y|\theta)\prod_{i=1}^{n} g(x_i|\theta)dx = p(\theta)\mathbb{E}[\hat{p}(y|\theta)] = p(\theta)p(y|\theta) \propto p(\theta|y) \tag{2.7}$$

Now I have illustrated how to use exact approximation likelihood to replace the true likelihood in MCMC. However, we should notice that when an unbiased estimator of the likelihood is used within a Metropolis–Hastings chain, it is necessary to trade off the number of Monte Carlo samples used to construct this estimator against the asymptotic variances of the averages computed under this chain. Using many Monte Carlo samples will typically result in Metropolis–Hastings averages with lower asymptotic variances than the corresponding averages that use fewer samples; however, the computing time required to construct the likelihood estimator increases with the number of samples. Schmon et al. [98] and Doucet et al. [28] discussed about how to implement Markov chain Monte Carlo method using an unbiased likelihood estimator efficiently.

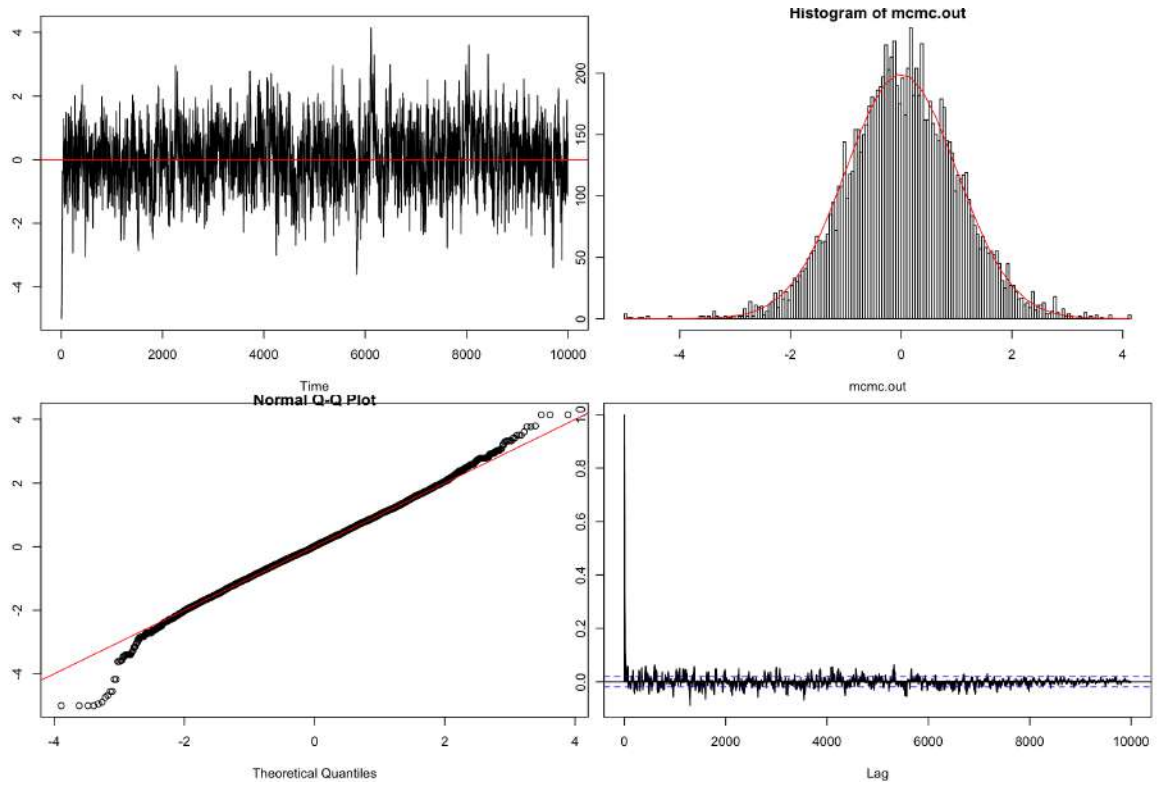

Figure 2.3: Pseudo-marginal Metropolis–Hastings with true distribution $N(0,1)$. The initial value is $x = -5$ and random walk step is 0.5. The top-left plot is for the trace-plot of Markov chain with theoretical mean value in red solid horizontal line. The top-right plot is for empirical distributions with theoretical distribution in red curve. Bottom-left plot is normal quantile-quantile plot and the bottom-right plot is autocorrelation plot.

Here are some examples employing Pseudo-marginal Metropolis–Hastings algorithm. If we want to sample from target distribution $N(0,1)$, we can directly use MCMC method described in Algorithm 4 when we can compute density of normal distribution exactly. However, in many cases, we can hardly access the likelihood as I discussed above. Assume we are unable to compute Normal distribution density, but we use different Monte-Carlo estimates instead.

Firstly, I present the result using the true likelihood, which is $N(0,1)$ distribution. As we can see from Figure 2.3, when we use the true distribution to calculate the likelihood in acceptance ratio, the algorithm performs very well. The Markov chain mixed well and the empirical distribution is very close to its theoretical density curve. Besides that, the Q-Q plot also shows the shape of the simulated distribution is very similar to its shape of its theoretical distribution, at the same time autocorrelation plot also suggests the Markov chain converge well with little autocorrelation after first 100 iterations.

However, sometimes we can not access to the true likelihood in acceptance ratio and we may use Monte Carlo estimates to replace the true likelihood. To be simplify, assume we can

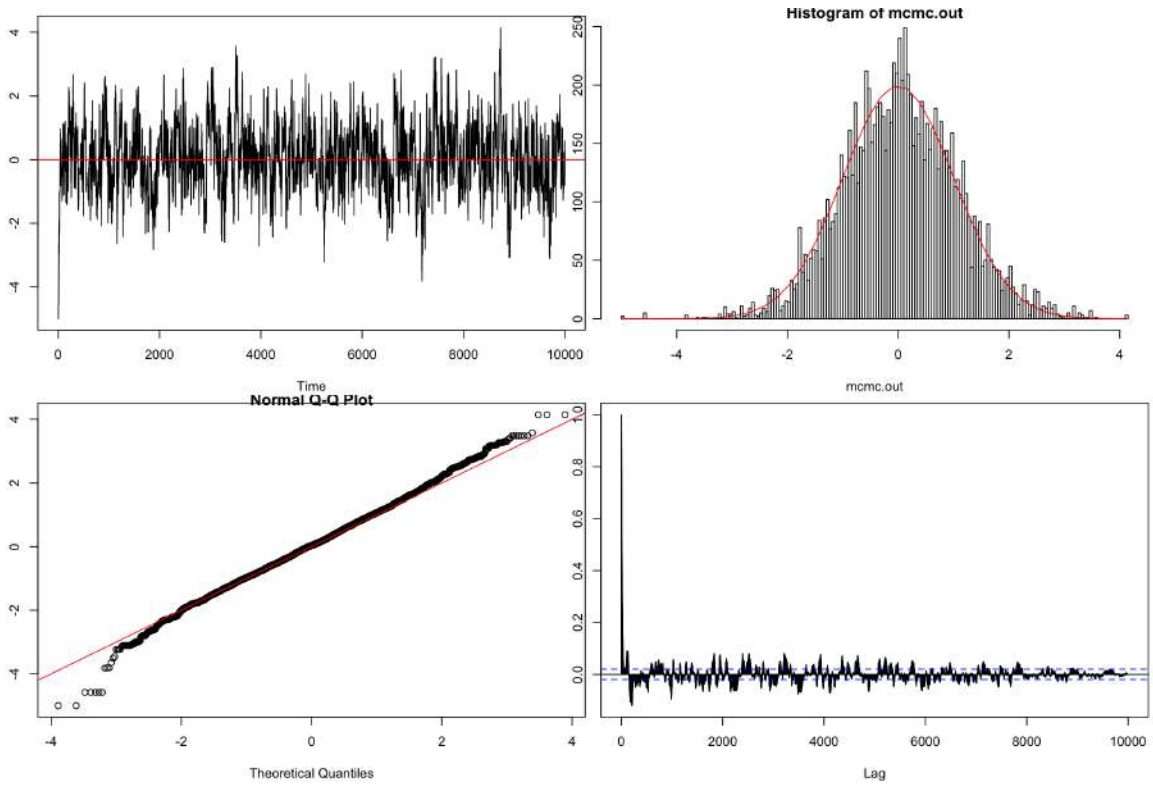

Figure 2.4: Pseudo-marginal Metropolis–Hastings with noisy distribution $N(0,1) \times \lambda$ where $\lambda \sim exp(10)$ and the theoretical distribution is $N(0,1)$. The initial value is $x = -5$ and random walk step is 0.5. The top-left plot is for the trace-plot of Markov chain with theoretical mean value in red solid horizontal line. The top-right plot is for empirical distributions with theoretical distribution in red curve. Bottom-left plot is normal quantile-quantile plot and the bottom-right plot is autocorrelation plot.

not calculate the distribution $N(0,1)$ but we can have an access to the function $N(0,1) \times \lambda$ where $\lambda \sim exp(10)$ and we use $N(0,1) \times \lambda$ to replace the true likelihood in pseudo-marginal Metropolis–Hastings algorithm. The results are presented in Figure 2.4. As we can see, the performance with noisy distribution $N(0,1) \times \lambda$ is also acceptable. $N(0,1) \times \lambda$ will result in a non-negative random quantity whose expectation is the expectation of the true likelihood function with a constant bias. As we use the Monte Carlo estimates on both numerator and denominator, so constant bias is also accepted in Pseudo-marginal Metropolis–Hastings. This lead to a similar result comparing with true distribution $N(0,1)$ used in Pseudo-marginal Metropolis–Hastings presented in Figure 2.4.

The last example is to use $0.5 \times N(-1, 0.25) + 0.5 \times N(1, 0.25)$ to replace the true likelihood function $N(0,1)$. Even though the mean of the noisy distribution is equal to the true distribution, the performance is unacceptable, which is presented in Figure 2.5. As we can see the noisy distribution is obviously not an unbiased estimate of the true likelihood function $N(0,1)$. Even

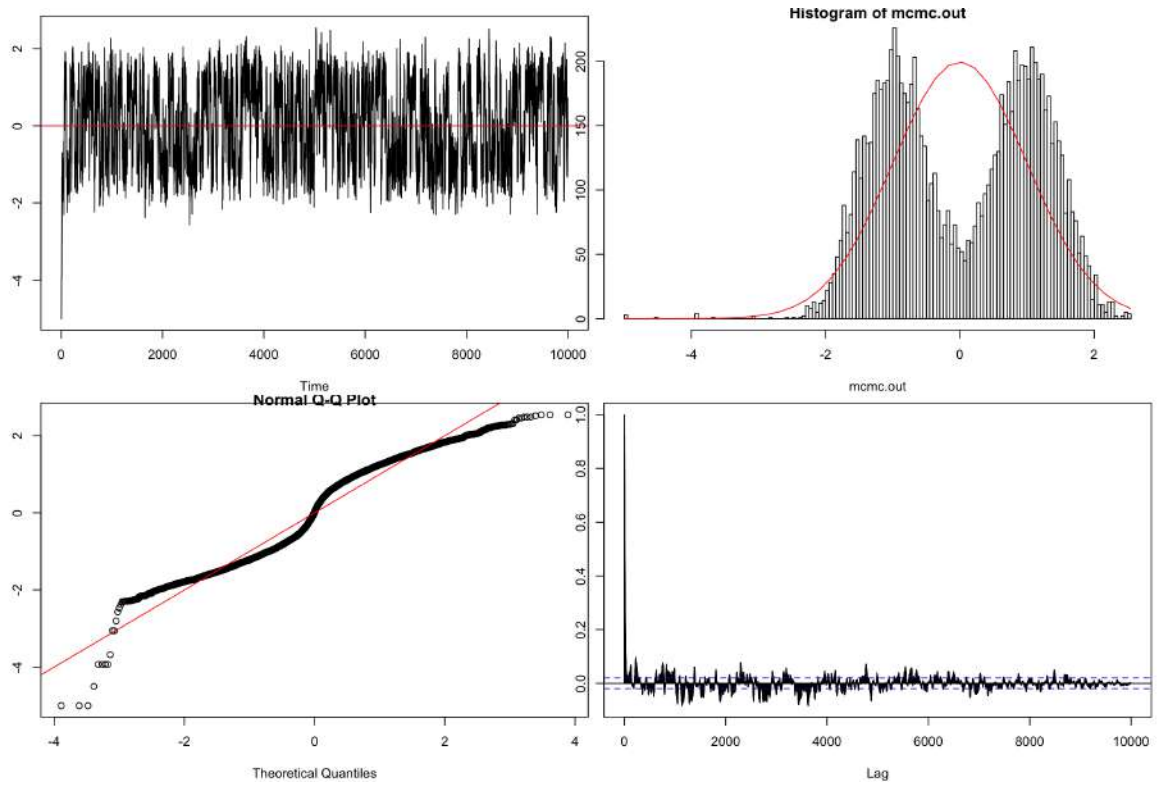

Figure 2.5: Pseudo-marginal Metropolis–Hastings with noisy distribution $0.5 \times N(-1, 0.25) + 0.5 \times N(1, 0.25)$ and the theoretical distribution is $N(0, 1)$. The initial value is $x = -5$ and random walk step is 0.5. The top-left plot is for the trace-plot of Markov chain with theoretical mean value in red solid horizontal line. The top-right plot is for empirical distributions with theoretical distribution in red curve. Bottom-left plot is normal quantile-quantile plot and the bottom-right plot is autocorrelation plot.

though the mean is the same, the trace-plot shows a good mixing, but the empirical distributions and quantile-quantile plot show the simulated samples are far away from the true distribution. It notices that we need to make sure we employing unbiased Monte Carlo estimates to replace the true likelihood when using the Pseudo-marginal Metropolis-Hastings algorithm, otherwise, it may lead to a very biased result. We can have unbiased Monte Carlo estimates to likelihood by using importance sampling[10] or particle filters [5], I will illustrate particle filters method in more detail in later chapters. In conclude, when using unbiased estimate of likelihood, the Pseudo-marginal Metropolis–Hastings is valid MCMC method.

2.2 Approximate Bayesian Computation (ABC)

2.2.1 Introduction of ABC

Approximate Bayesian computation (ABC) is a family of methods for approximate inference, used when likelihoods are impossible or impractical to evaluate numerically but simulating datasets from the model of interest is straightforward. ABC can be viewed as a nearest neighbours method. It simulates datasets given various parameter values, and finds the closest matches, in some sense, to the observed dataset. The corresponding parameters are used as the basis for inference. Various Monte Carlo methods have been adapted to implement this idea, including rejection sampling[15], ABC-MCMC[76] and sequential Monte Carlo [104]. In this section, I will briefly introduce the ABC concept and some ABC techniques used in later chapters.

In Bayesian inference, the prior distribution, $\pi(\theta)$, reflects one's prior belief on parameters $\theta \in \vartheta$. Observing data $y_{obs} \in Y$ update this prior knowledge by likelihood function $\pi(y_{obs}|\theta)$. The target of Bayesian inference is the conditional distribution, $\pi(\theta|y_{obs})$, known as the posterior distribution. By using Bayes' Theorem, it can be expressed as follow,

$$\pi(\theta|y_{obs}) = \frac{p(y_{obs}|\theta)\pi(\theta)}{\int_{\vartheta} p(y_{obs}|\theta)\pi(\theta)d\theta} \tag{2.8}$$

For tractable cases, the likelihood function, $\pi(y_{obs}|\theta)$, can be analytically explicit and the posterior distribution can be calculated directly by Equation 2.8 or by using a simulation method e.g. Markov chain Monte Carlo. However with the complexity of the model involved, the likelihood function is often intractable, and thus we can not evaluate Equation 2.8 and figure out the posterior distribution analytically. Actually, such cases are even more common in population genetics, so we need to turn to the use of likelihood-free algorithms and approximate Bayesian computation (ABC) is one of such classes algorithms.

A statistic is biased if it is calculated in such a way that it is systematically different from the population parameter being estimated. If $E(T) = \theta + bias(\theta)$, the $bias(\theta)$ is the bias of statistic$T$ where $E(T)$ is the expectation value of statistic$T$. If $bias(\theta) = 0$, then T is unbiased estimator of parameter θ.

A precursor ABC algorithm was introduced by Rubin[96] in 1984. The original idea is to simulate data, denoted simulated data as $\mathbf{x}$. Accept with parameters which generate data that can exactly match with observed data $\mathbf{y}$ with a tolerance ϵ. Thus we can have an approximate posterior distribution denoted $p_\epsilon(\theta, \mathbf{x}|\mathbf{y})$, where

$$p_\epsilon(\theta, \mathbf{x}|\mathbf{y}) \propto \pi(\theta)p(\mathbf{x}|\theta)\mathbb{I}\{d(\mathbf{x}, \mathbf{y}) < \epsilon\} \tag{2.9}$$

$d(\mathbf{x},\mathbf{y})$ is the distance between $\mathbf{x}$ and $\mathbf{y}$ and $\mathbb{I}(\cdot)$ is the indicator function,

$$\mathbb{I}\{z\} = \begin{cases} 1, & \text{if z is true} \\ 0, & \text{otherwise} \end{cases} \tag{2.10}$$

As we can see in Equation2.9, the smaller the ϵ is, the closer the ABC posterior distribution to the true posterior distribution. But as ϵ goes to zero, the running time of making efficient inferences may become unacceptable.

Here is a toy example. Consider a Normal sample with N observations $\mathbf{y} := (y_1, y_2, \cdots, y_N)$, which are independently draw from a Normal distribution $N(\mu,\sigma^2)$ with unknown mean μ and unknown variance σ^2. We want to compute the posterior distribution $p(\theta|\mathbf{y})$ to infer the parameters $\theta := \left(\mu,\sigma^2\right) \in \Theta_{\Re\times\Re}$ based on dataset $\mathbf{y}$. Here I choose to use the conjugate Normal-inverse-Gamma (NIG) prior distribution on $\theta := \left(\mu,\sigma^2\right)$ which is given as,

$$p(\mu,\sigma^2) = NIG(m_0,v_0,a_0,b_0) = N(\mu|m_0,\sigma^2 v_0)IG(\sigma^2|a_0,b_0) \tag{2.11}$$

Thus we have the prior distribution with hyper-parameters given as,

$$\sigma^2 \sim Inv-Gamma(a_0,b_0) \tag{2.12}$$

$$\mu|\sigma^2 \sim N(m_0,\sigma^2 v_0) \tag{2.13}$$

where (m_0,v_0,a_0,b_0) are the hyper-parameters

As Normal-inverse-Gamma distribution is one of the conjugate distributions for a sample from the Normal distribution, the posterior distribution also follows the Normal-inverse-Gamma distribution which is,

$$p(\mu,\sigma^2|\mathbf{y}) = NIG(m_N,v_N,a_N,b_N) = N(\mu|m_N,\sigma^2 v_N)IG(\sigma^2|a_N,b_N) \tag{2.14}$$

where

$$v_N = \left(v_0^{-1} + N\right)^{-1} \tag{2.15}$$

$$m_N = v_N \times \left(v_0^{-1} m_0 + N\bar{y}\right) \tag{2.16}$$

$$a_N = a_0 + \frac{N}{2} \tag{2.17}$$

$$b_N = b_0 + \frac{m_0^2 v_0^{-1} + \sum_i^N y_i^2 - m_N^2 v_N^{-1}}{2} \tag{2.18}$$

In my example, I choose the value for the hyper-parameters $(m_0, v_0, a_0, b_0) := (1, 8, 3, 8)$ and the observation $\mathbf{y}$ is independently simulated from the Normal distribution $N(1, 4)$. This example is very simple, as we know the Normal-inverse-Gamma distribution is conjugate to Normal distributions, we can have the theoretical value of parameters of the posterior distribution very easily. Then we can you the different ABC parameter estimators to compare with its theoretical posterior parameter value to conclude how different ABC scheme performs. The limitation for this example is also obvious, I draw sample observation from the Normal distribution once and reuse them in different ABC test. It lost generality, but the main point here is to present how ABC performs using different ABC techniques with different set of summary statistics.

Then let us follow the original rejection algorithm described in Rubin[96] with the different combinations of the tolerance ϵ and the number of the observations N and calculate the average number of the simulations needed to generate one accepted rejection ABC sample. The result is presented in the Table 2.1.

Table 2.1: The comparison of the average number of simulations needed to generate 1 accepted rejection ABC sample, the maximum number of simulations used here is 10^7. The Inf in the table means there does not exist any sample within the accepted tolerance distance among 10^7 simulations.

ϵ	50	5	0.5
N = 3	1.000	6.146	6285.355
N = 10	1.004	397.994	Inf
N = 100	4.698	Inf	Inf
N = 1000	Inf	Inf	Inf

There are two important drawbacks in this precursor ABC algorithm. Firstly, we need a good scheme to balance tolerance ϵ with computational expense as I mentioned above. The second thing is when the dimension of the dataset is large, it is hard to be accepted, which lead the algorithm to an ineffective result. For example, the observation $y_{obs} = (1, 2, 3)$ and there are two simulated dataset which are $x_1 = (3, 2, 1)$ and $x_2 = (1, 2, 3)$. In our example, the observation and simulated dataset are independent and identically draw from distribution, which mean they should be exchangeable sequence of random variables. An exchangeable sequence of random variables is a sequence $(X_1, X_2, X_3, \cdots)$ whose joint probability distribution does not change when the positions in the sequence in which finitely many of them appear are altered. However, when we following Equation2.9 to calculate the distance between the observation and simulated dataset, the $d(x_1, y)$

will be obviously bigger than $d(x_2, y)$. It lead to a problem that the exchangeable random variables result in different distance calculation result and when the number of observation is increasing, it will be hard to accept the simulated dataset, even though they are simulated from the distribution with same parameter of observations.

This drawback has encouraged people to figure out some method to reduce the dimension of observations and simulation data instead of comparing the observation with the a mount of simulation sample point by point.

2.2.2 Summary Statistics in ABC

Besides the likelihood-free intuitive idea, Pritchard et al[91] introduced summary statistics into ABC following Weiss and von Haeseler [121]. It is a great innovation that summary statistics largely reduced the dimension of simulated data and observations. This made the ABC framework possible and efficient to compare the distance between simulation and observations. For example, in Pritchard et al[91], they collected a dataset with 445 male humans around the world. In each sample, they tried to use the variation across 8 Y-chromosome microsatellite loci to uncover the genetic structure and the most recent common ancestor(MRCA) under a growth model which they believed would be more suitable to explain the demographic history. To make comparison between the observations and simulated points in an efficient way, Pritchard et al[91] used 3 summary statistics: the mean-variance in repeat numbers, the mean effective heterozygosity and the number of distinct haplotypes. They thought the 3 summary statistics are informative and closely related to mutation rate, population size and MRCA respectively.

By using summary statistics, a simple and low dimension comparison between simulated data and observed data can be constructed, and using that, we can accept parameters which generate the summary statistics that are within a small distance from the summary statistics of the observation.

$$p_\epsilon(\theta, s(\mathbf{x})|s(\mathbf{y})) \propto \pi(\theta)p(\mathbf{x}|\theta)\mathbb{I}\{\rho(s(\mathbf{x}), s(\mathbf{y})) < \epsilon\} \tag{2.19}$$

where $s(\mathbf{x}) = s_1(\mathbf{x}), \cdots, s_p(\mathbf{x})$ denotes the summary statistics of data set $\mathbf{x}$, and the dimension of this set of summary statistics is p, and in the example above, $p = 3$. $\rho(\cdot, \cdot)$ is a distance metric on summary statistics, which can be written as $\|s(\mathbf{x}) - s(\mathbf{y})\|$. The commonest distance metric used in ABC is the Euclidean distance, but there are alternative distance metrics can be used in ABC, and for example, Pritchard et al[91] used the Chebyshev distance metric.

By using of summary statistics, we can use some permutation invariant summary statistics, for example, mean, variance and so on, to summary exchangeable sequence of random variables. As no matter what the order of simulated data, the expectation of $E(x_1 = (1,2,3)) = 2$ is equal to the $E(x_2 = (3,2,1)) = 2$. We can regard the computation of summary statistics as a set of mappings from a high dimension to a low dimension. Typically information is lost, but, with enough of

these low dimensional summaries much of the information in the high-dimensional data may be captured.

Here we can follow the toy example in section 2.1, picking the number of observations $\mathbf{N} = 3, 10, 100, 1000$, and choose different sets of summary statistics to compare the performance with different bandwidth. For simplicity, I chose 1000 accepted ABC samples from $\pi_{ABC}(\theta|\mathbf{y})$ to draw a histogram. For comparison, I used $S_1(\mathbf{y}) := \frac{1}{N}\sum_{i=1}^{N} y_i$, $S_2(\mathbf{y}) := Var(\mathbf{Y})$, $S_3(\mathbf{y}) := Median(\mathbf{y})$, $S_4(\mathbf{y}) := \sqrt{Var(\mathbf{y})}$ to summarise the observation and simulations. We want to calculate the posterior distribution $p(\mu, \sigma^2|\mathbf{y})$ and here we can also calculate the theoretical posterior distribution by Equation 2.14 to Equation 2.18. I list the summary statistics of the observations in Table 2.2 and the calculating results of the theoretical posterior distribution parameters in Table 2.3.

Table 2.2: The summary statistics value of the observation data with the different number of the observation $N = 3, 10, 100, 1000$.

summary statistic	mean	variance	median	standard deviation
$\mathbf{N} = 3$	2.070	0.552	2.309	0.74
$\mathbf{N} = 10$	0.129	1.88	0.051	1.37
$\mathbf{N} = 100$	0.88	4.83	0.584	2.19
$\mathbf{N} = 1000$	1.021	4.22	1.006	2.05

Table 2.3: The theoretical posterior distribution parameters value with the different number of the observation $N = 3, 10, 100, 1000$. The posterior mean of σ^2 in table is equal to the value $\frac{b_N}{a_N - 1}$

$parameter$	v_N	m_N	a_N	b_N	posterior mean of σ^2
$\mathbf{N} = 3$	0.320	2.027	4.5	8.62	2.463
$\mathbf{N} = 10$	0.098	0.139	8	16.5453	2.363
$\mathbf{N} = 100$	0.009	0.888	53	247.55	4.76
$\mathbf{N} = 1000$	0.001	1.021	503	2117.546	4.218

For this example, I randomly draw a sample with total observation is $\mathbf{N} = 100$, the summary statistics for this observed data are $S(\cdot) := (S_1(\mathbf{y}), S_2(\mathbf{y}), S_3(\mathbf{y}), S_4(\mathbf{y}))$ which is $S(\mathbf{y}) = (0.888, 4.839, 0.584, 2.199)$ for this simulated observation data. The theoretical posterior distribution for μ and σ^2 is $p(\mu|\mathbf{y}; \sigma^2) \sim N(0.888, 0.009\sigma^2)$ and $p(\sigma^2|\mathbf{y}) \sim Inv - Gamma(53, 247.559)$ respectively. I present the output for the example $\mathbf{N} = 100$ in Figure 2.6. The marginal distribution for σ^2 is the Inv-Gamma distribution given as above, but there is an integral on σ^2 needed to calculate the marginal distribution of μ. Following the calculation process given in Gelman et al. [42], we can find the marginal distribution of μ is given as,

$$(\mu - m_N)\sqrt{\frac{a_N}{b_N v_N}} \sim t_{2a_N}$$

where the t_{2a_N} is Student's t-distribution with $2a_N$ degree freedom.

In Figure 2.6, the number of total simulations I use is 10^7 and the number of accepted samples is 41641 and 3768 for $\epsilon = 0.1$ and $\epsilon = 0.03$ respectively. By using the mean and variance as

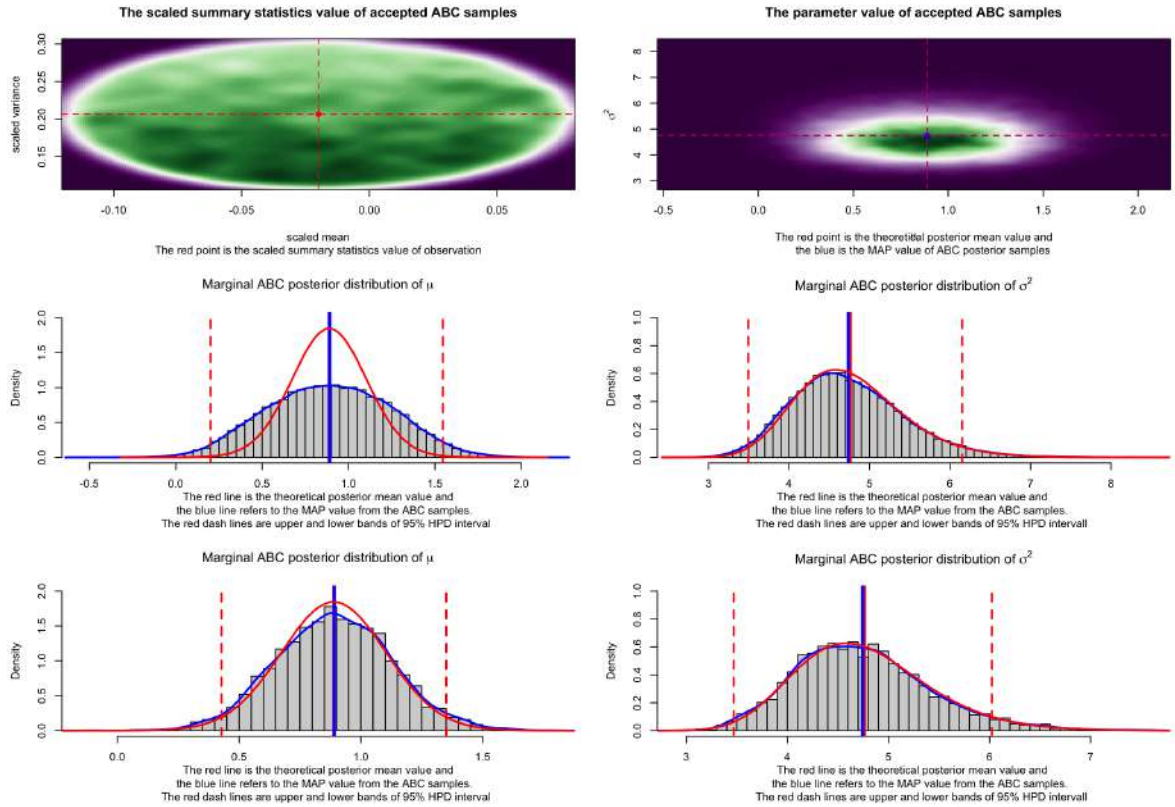

Figure 2.6: Output from ABC based on N=100 samples with summary statistics $S(\cdot) = \{S_1 = mean(\cdot), S_2 = var(\cdot)\}$. The top left figure is the joint distribution of scaled summary statistics of the accepted simulation samples. The red point is the scaled summary statistics of the observation data. The top right figure is the joint distribution of the accepted ABC sample parameters with $\epsilon = 0.1$. The blue point in the figure is the posterior distribution mean of μ and σ^2 from ABC samples and the red points is the theoretical posterior mean of the μ and σ^2. The middle left and middle right figures are the marginal posterior distributions for μ and σ^2 with $\epsilon = 0.1$. The bottom left and bottom right figures are the marginal posterior distributions for μ and σ^2 with $\epsilon = 0.03$. In those histogram, the blue vertical lines are the ABC posterior mean of μ and σ^2 from the accepted ABC samples and the red solid vertical lines are the theoretical posterior means of μ and σ^2. The red dotted vertical lines are the 95% HPD intervals of the theoretical posterior distributions of μ and σ^2. The blue curve is the smooth density curve from the accepted ABC samples of μ and σ^2 and the red curve is the theoretical marginal posterior density curve of μ and σ^2. The total number of the simulation here is 10^7.

summary statistics, we can find an effective posterior ABC samples for the number of observations $\mathbf{N} = 100$. The ABC posterior distribution mean of μ is very close to the theoretical posterior mean value, the similar case to the σ^2. However, due to the ϵ is not small enough, the empirical distributions marginal distribution of μ is broader than its theoretical posterior distribution density curve when $\epsilon = 0.1$. This issue turns to be better when we decrease the tolerance ϵ and

I present the result from same simulation data with $\epsilon = 0.03$, as we can see the ABC posterior empirical distributions density curve is much close to its theoretical posterior density curve than before.

Table 2.4: The ABC posterior mean of μ for different number of observations $N = 3, 10, 100, 1000$ and different sets of summary statistics. The maximum number of simulations used here is 10^6. The tolerance is $\epsilon = 0.1$

μ	S_1, S_2	S_1, S_4	S_3, S_2	S_3, S_4	theoretical posterior mean
N = 3	2.038	2.037	2.087	2.066	2.0275
N = 10	0.142	0.124	0.137	0.1319	0.139
N = 100	0.873	0.881	0.850	0.884	0.888
N = 1000	1.021	1.028	1.031	1.026	1.021

Table 2.5: The ABC posterior mean of σ^2 for different number of observations $N = 3, 10, 100, 1000$ and different sets of summary statistics. The maximum number of simulations used here is 10^6. The tolerance is $\epsilon = 0.1$

σ^2	S_1, S_2	S_1, S_4	S_3, S_2	S_3, S_4	theoretical posterior mean
N = 3	2.508	2.667	2.450	2.641	2.463
N = 10	2.312	2.46	2.316	2.47	2.36
N = 100	4.714	4.00	4.721	4.00	4.760
N = 1000	4.191	3.683	4.192	3.65	4.21

From the Table 2.4 and Table 2.5, we can find when the number of the observations **N** is small, the results of ABC samples are very sensitive to its observations and may result in a biased result. However, as I illustrate in section 2.1, without using summary statistics, it is impossible to use a large number of observations to make a comparison between the simulated data and the observations. The use of summary statistics in ABC allows us to make an inference based on a large data set. The more observations are involved into analysis, the closer the ABC posterior mean of the accepted samples approaches to its theoretical posterior mean. The bias of posterior parameter estimators are expected to be decreasing as the number of observation increasing since the more observation we have, the better knowledge we extract from the target population distribution.

Besides the number of observations N, the choice of summary statistics also affects the performance of ABC a lot. From Table 2.4 and Table 2.5 we are also able to find when the choice of summary statistics is different, the performance of ABC changes. I present the joint distribution of the accepted ABC samples parameters using different summary statistics in Figure 2.7 to illustrate how the choice of summary statistics affect the ABC posterior distribution on μ and σ^2. The estimate of parameter μ is close to its theoretical posterior mean value for all different sets of summary statistics in all trials. Comparing those figures and results in Table 2.4 and Table 2.5, the change from using mean as one of the summary statistics to median makes nearly no difference irrespective of the other summary statistic is variance or standard deviation. However,

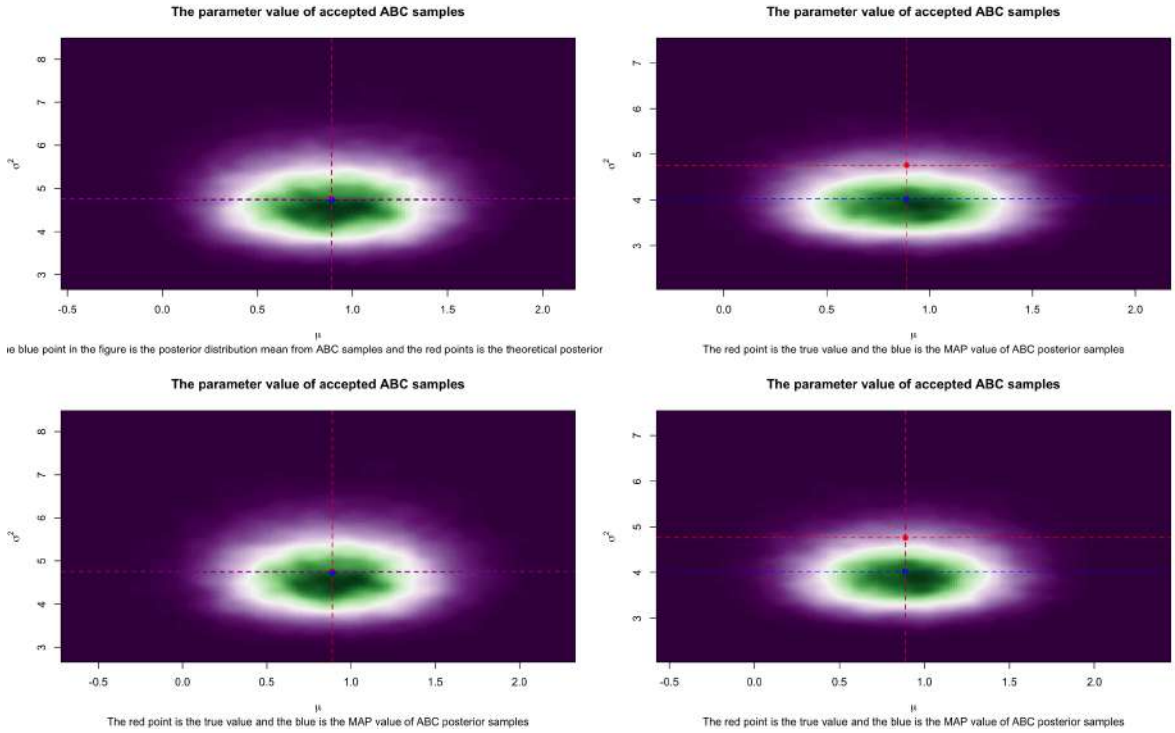

Figure 2.7: The joint distribution of the accepted ABC samples parameters using different summary statistics based on observation N=100 with different sets of summary statistics. The set of summary statistics used is $S^{(1)}(\cdot) = \{S_1 = mean(\cdot), S_2 = var(\cdot)\}$ (the top left), $S^{(2)}(\cdot) = \{S_1 = mean(\cdot), S_4 = \sqrt{var(\cdot)}\}$ (the top right), $S^{(3)}(\cdot) = \{S_3 = median(\cdot), S_2 = var(\cdot)\}$ (the bottom left) and $S^{(4)}(\cdot) = \{S_3 = median(\cdot), S_4 = \sqrt{var(\cdot)}\}$ (the bottom right). The tolerance is still $\epsilon = 0.1$ and the total number of the simulation is 10^7. The blue point in the figure is the ABC posterior mean value of μ and σ^2 and the red points is the theoretical posterior mean of the μ and σ^2.

the change from using variance as one of the summary statistics to standard deviation leads to bias, especially in estimating σ^2. The ABC posterior mean value of σ^2 based using variance as summary statistics is 4.71 and 4.72 for using mean and median respectively, and the theoretical posterior mean of σ^2 based on the observation data $N = 100$ is 4.76. When using standard deviation as summary statistics, the MAP estimate of σ^2 is 4.000 and 4.001 for using mean and median respectively, which are far away from the theoretical posterior value of σ^2. When the sample size increases to $N = 1000$, the bias becomes more significant for the ABC posterior mean of σ^2 listed in the Table 2.4 and Table 2.5. The importance of this example here is to illustrate that different sets of summary statistics will lead to different estimate, so choosing a suitable summary statistics is very important in ABC.

In addition to the summary statistics I introduce above, there is a criterion issue that needs to be noted. In the above example, I chose to use a 'rejection' intuitive idea, i.e., $\rho(s(\mathbf{x}), s(\mathbf{y})) \leq \epsilon$. It is convenient to implement this in computer programming within an "if-else" statement. However,

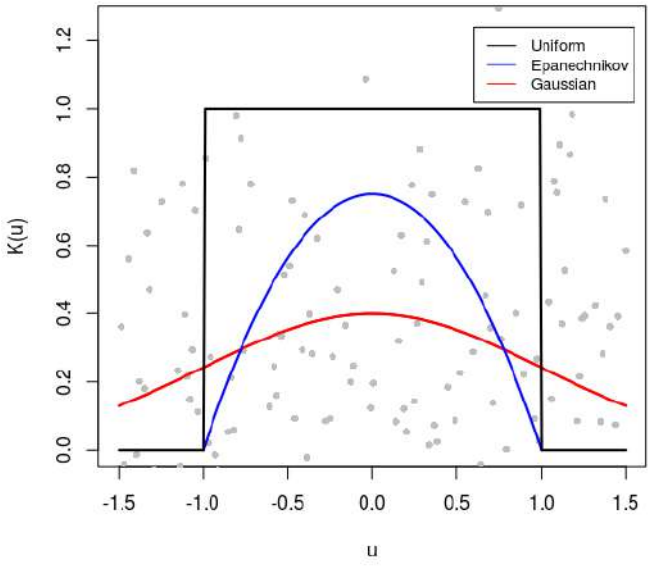

Figure 2.8: Different kernel function performs

such a rejection idea is wasteful of information as it does not discriminate between those ABC samples θ, for which the associated dataset $\mathbf{y}$ is exactly equivalent to the observation $\mathbf{y}_{obs}$ i.e. $\rho(s(\mathbf{x}),s(\mathbf{y}))=0$, and samples θ, for which the associated dataset $\mathbf{y}$ is far away from $\mathbf{y}_{obs}$ i.e. $\rho(s(\mathbf{x}),s(\mathbf{y}))=\epsilon$. This issue can be solved by replacing $\rho(s(\mathbf{x}),s(\mathbf{y}))\leq\epsilon$ with a smoothing kernel function $K_\epsilon(\cdot)$, where ϵ is the tolerance term. There are some conditions needed when introducing the kernel function $K_\epsilon(\cdot)$. If the distance is u, then $K_\epsilon(u)\geq 0$, $\int K_\epsilon(u)du=1$, $\int uK_\epsilon(u)du=0$ and $\int u^2K_\epsilon(u)du<\infty$.

$$p_\epsilon(\theta,s(\mathbf{x})|s(\mathbf{y}))\propto \pi(\theta)p(\mathbf{x}|\theta)K_\epsilon(\|s(\mathbf{x})-s(\mathbf{y})\|) \tag{2.20}$$

If we directly use a distance comparison between simulation and observation summary statistics, it means we use a Uniform kernel as in Equation 2.10. Besides the Uniform kernel, there are some other common kernels that are often involved in the ABC framework, shown in Figure 2.8, such as Epanechnikov and Gaussian kernel. The Epanechnikov kernel function is, $K_E(u)=\frac{3}{4}\times(1-u^2)$ where $|u|\leqslant 1$ and the Gaussian kernel is, $K_G(u)=\frac{1}{\sqrt{2\pi}}exp(-\frac{1}{2}u^2)$. The different choices of kernel function will affect the performance of ABC and reshape the accepted region.

Now by using summary statistics and the idea above, we can present the standard Rejection ABC that was used by Pritchard et al[91].

Algorithm 4 standard Rejection ABC

Initialization: choose suitable tolerance ϵ and summary statistics $S(\cdot)$
1. Sample parameter from prior $\theta_i \sim \pi(\theta)$
2. Simulate $S(\mathbf{x_i})$ from generative model $p(\mathbf{x_i}|\theta_i)$
3. Reject with probability proportion to kernel density $K_\epsilon(\|s(\mathbf{x_i}) - s(\mathbf{y})\|)$
4. Repeat step 1-3 until M acceptances are obtained

2.2.3 Sufficient Summary Statistics

From Equation 2.9, we can easily find that when ϵ goes to 0, the ABC posterior will tend to the true posterior distribution, which is

$$p_\epsilon(\theta, \mathbf{x}|\mathbf{y}) \to p(\theta|\mathbf{y}) \quad \text{as} \quad \epsilon \to 0 \tag{2.21}$$

But in the case of summary statistics, this situation only holds when the set of summary statistic $S(\cdot)$ is 'sufficient' for the parameter space θ. It is equivalent that the summary statistics $S(\cdot)$ is conditionally independent of parameter θ. That means all the information of parameter θ contained in data is captured by the summary statistic $S(\cdot)$, which can be presented as

$$p(\mathbf{x}|s(\mathbf{x}), \theta) = p(\mathbf{x}|s(\mathbf{x})) \tag{2.22}$$

Practically, the Equation 2.22 can be presented following Fisher-Neyman factorization Theorem [111], which shows the sufficiency of the summary statistic by writing the probability density function into two factorized non-negative function,

$$f(\mathbf{x}|\theta) = h(\mathbf{x})g(S(\mathbf{x})|\theta) \tag{2.23}$$

where both g and h are non-negative function. For example, if we continue our Normal distribution example that N observations $\mathbf{y} := (y_1, y_2, \cdots, y_N)$ are independently drawn from the Normal distribution $N(\mu, \sigma^2)$, its probability density function is,

$$f(\mathbf{y}|\theta) = \prod_{i=1}^{N} \frac{1}{\sqrt{2\pi\sigma^2}} exp\left(-\frac{(y-\mu)^2}{2\sigma^2}\right) \tag{2.24}$$

which can be written as,

$$f(\mathbf{y}|\theta) = (2\pi\sigma^2)^{-\frac{n}{2}} exp\left(-\frac{1}{2\sigma^2}\sum_{i=1}^{N}(y_i - \bar{y})\right) exp\left(-\frac{n}{2\sigma^2}(\mu - \bar{y})\right) \tag{2.25}$$

We can regard $h(\mathbf{y}) = (2\pi\sigma^2)^{-\frac{n}{2}} exp\left(-\frac{1}{2\sigma^2}\sum_{i=1}^{N}(y_i - \bar{y})\right)$ and $g(S(\mathbf{y})|\mu) = exp\left(-\frac{n}{2\sigma^2}(\mu - \bar{y})\right)$. As we can see, the $h(\mathbf{y})$ is only depend on the observation itself and $g(S(\mathbf{y})|\mu)$ is dependent on the

observations only through summary statistics function $S(\mathbf{y}) := \frac{1}{N}\sum_{i=1}^{N} y_i = \bar{y}$. By using Fisher-Neyman Theorem in equation 2.23, we can find the $S(\mathbf{y}) := \frac{1}{N}\sum_{i=1}^{N} y_i$ is a sufficient statistic for parameter μ. Similarly, if we substitute $s^2 = \frac{1}{n-1}\sum_{n=1}^{N}(y_i - \bar{y})$ into equation 2.25, we can have,

$$f(\mathbf{y}|\theta) = (2\pi\sigma^2)^{-\frac{n}{2}} exp\left(-\frac{n-1}{2\sigma^2}s^2\right) exp\left(-\frac{n}{2\sigma^2}(\mu-\bar{y})\right) \tag{2.26}$$

The Fisher–Neyman factorization theorem in equation 2.23 still holds for parameter σ^2 through summary statistics $s^2 = \frac{1}{n-1}\sum_{n=1}^{N}(y_i - \bar{y})$, which implies the set of summary statistics $S(\cdot) := \{S_1 = mean(\cdot), S_2 = var(\cdot)\}$ is a jointly sufficient summary statistic for parameter $\theta := (\mu, \sigma^2)$.

The choice of a sufficient summary statistics is a key problem in the ABC framework and significantly affects the performance of ABC as I present in section 2.2.2. We need to choose informative summary statistics which can capture as much as possible information in the data set. Based on prior knowledge, we can subjectively choose some summary statistics which are closely related to some key parameters as in Pritchard et al[91]. As ABC has developed, there are now two main approaches to figure out how to have a good set of summary statistics: one is to find the optimal subsets of summary statistics[59][87]; another method is to find an optimal projection of summary statistics into a lower dimension[13][35]. Here I will focus on an illustration on the projection method which is much more relevant to my later chapters.

2.2.3.1 Minimising Mean Squared Error

Fearnhead and Prangle[35] pointed out an idea that comparing with a full estimation of the posterior distribution, it is easy and often sufficient to make a precise point estimate from the posterior distribution. We can define the quadratic loss function of the point estimate value $\hat{\theta}$ and true parameter value θ with a positive definite matrix A as,

$$\mathscr{L}(\theta, \hat{\theta}; A) = (\theta - \hat{\theta})A(\theta - \hat{\theta})^T \tag{2.27}$$

Then we consider the expectation of this squared loss function given the observation $\mathbf{y}$, which is

$$\mathbb{E}(\mathscr{L}(\theta, \hat{\theta}; A)|\mathbf{y}) = \int (\theta - \hat{\theta})A(\theta - \hat{\theta})^T p(\theta|\mathbf{y})d\theta \tag{2.28}$$

We can obtain the minimum value of the expectation quadratic loss function given A is a positive definite matrix and calculate the minimum value by differentiating both side with respect to θ and set it equal to 0, which is,

$$\frac{d}{d\theta}\mathbb{E}(\mathscr{L}(\theta, \hat{\theta}; A)|\mathbf{y}) = \frac{d}{d\theta}\int (\theta - \hat{\theta})A(\theta - \hat{\theta})^T p(\theta|\mathbf{y})d\theta = 0 \tag{2.29}$$

Then we can find the derivative of the expected quadratic loss function given observations $\mathbf{y}$ is equal to $2A\int\left(\hat{\theta}-\theta\right)p(\theta|\mathbf{y})d\theta$, so we have,

$$0 = 2A\int\left(\hat{\theta}-\theta\right)p(\theta|\mathbf{y})d\theta = \int\theta p(\theta|\mathbf{y})d\theta - \int\hat{\theta}p(\theta|\mathbf{y})d\theta = \mathbb{E}(\theta|\mathbf{y}) - \hat{\theta} \tag{2.30}$$

By minimizing the expected quadratic loss function, we have that the best estimate of parameter θ is the posterior mean of it, i.e., $\hat{\theta} = \mathbb{E}(\theta|\mathbf{y})$. It marginally implies the posterior mean for each parameter is a good and sufficient summary statistic [35]. However, the quantity is not accessible because we can not know the true value of the posterior mean. But it leads to the idea that a suitable choice of summary statistics should be an efficient estimate of the posterior mean.

2.2.3.2 Semi-automatic ABC

Besides the idea of using the posterior mean as a summary statistic, Fearnhead and Prangle also imply that the optimal number of summary statistics involved in the ABC algorithm should be equal to the number of parameters that we are interested in[35]. A later study on asymptotic behaviour of ABC from Li and Fearnhead shows that when the number of summary statistics, denoted N_s, is larger than the number of parameters involved, N_p, there always exist a projection of summary statistics down to N_p dimensions with a lower or equal asymptotic variance comparing with the use of N_s dimensional summary statistics[67].

Fearnhead and Prangle [35] also suggested a very practical method which they called semi-automatic ABC to construct a suitable summary statistic for each parameter. The basic idea is to use simulated sets of parameter value and data to estimate summary statistics. There are many approaches to make those estimates, in Fearnhead and Prangle [35], they test to use canonical correlation analysis and lasso method, but in general, the most practical linear regression method with appropriate functions of the data as predictor is both simpler and worked better than the method listed above. We can use linear regression to find the estimate of the posterior mean in our example directly. They mentioned that there existed many estimation methods, and they chose to use linear regression because it was efficient and with a lower computational cost. By sampling from the joint distribution of parameters and summary statistics $p(\theta_i, S(\mathbf{x}_i)$, where $i = 1,\cdots,M$, the linear model is:

$$\theta_i^T = (S(\mathbf{x}_i))^T\beta + \epsilon_i^T \tag{2.31}$$

where

$$\theta_i = [\theta_{i,1},\cdots,\theta_{i,N_p}]^T,$$

$$S(\mathbf{x}_i) = [1, S_1(\mathbf{x}_i),\cdots,S_{N_s}(\mathbf{x}_i)]^T,$$

$$\epsilon_i = [\epsilon_{i,1},\cdots,\epsilon_{i,N_p}]^T$$

The β is matrix of regression coefficient with $(N_s+1)\times N_p$ dimentions, and this β can be substituted with its ordinary least squares estimate $\hat{\beta}$ when given samples from the joint distribution $p(\theta_i, S(\mathbf{x}_i)$. An algorithm with Semi-automatic ABC summary statistic can be shown as,

Algorithm 5 Rejection ABC with Semi-automatic ABC summary statistic

Initialization: choose suitable tolerance ϵ and summary statistics $S(\cdot)$
1. Sample N parameters from prior $\theta_i \sim \pi(\theta)$
2. Simulate $S(\mathbf{x}_{1,\cdots,N})$ from generative model $p(\mathbf{x_i}|\theta_i)$
3. Center and scale $S(\mathbf{x}_{1,\cdots,N})$ to have zero mean and unit variance
4. For each parameter $\theta_{\cdot,j}$, obtain $(\hat{\beta}^j)^T S(\mathbf{x})$ as projected summary statistic
5. Reject with probability proportion to kernel density $K_\epsilon(\left\|(\hat{\beta}^j)^T S(\mathbf{x}_i) - (\hat{\beta}^j)^T(1, s(\mathbf{y}))\right\|)$
6. Repeat step 1-5 until M acceptances are obtained

Here I continue to use my former Normal distribution example with the same original summary statistics $S(\cdot) := \left(mean(\cdot), Var(\cdot), median(\cdot), \sqrt{Var(\cdot)}\right)$. Then we use the regression method following the idea illustrated in Semi-automatic ABC summary statistic algorithm to calculate the projected summary statistic corresponding to different parameter $\theta := (\mu, \sigma^2)$ and present the weights from fitted regressions in Table 2.6, Table 2.7 and Table 2.8.

Table 2.6: The weights from fitted regressions of μ and σ with S_1, S_2

$(\hat{\beta}^j)^T$	$S_1 := mean(\cdot)$	$S_2 := Var(\cdot)$	$S_3 := Median(\cdot)$	$S_4 := \sqrt{Var(\cdot)}$
μ	5.645	$6.303e^{-5}$		
σ^2	0.006	3.83		

Table 2.7: The weights from fitted regressions of μ and σ with S_3, S_4

$(\hat{\beta}^j)^T$	$S_1 := mean(\cdot)$	$S_2 := Var(\cdot)$	$S_3 := Median(\cdot)$	$S_4 := \sqrt{Var(\cdot)}$
μ			5.643	0.0001
σ^2			-0.003	3.563

Table 2.8: The weights from fitted regressions of μ and σ with S_1, S_2, S_3, S_4

$(\hat{\beta}^j)^T$	$S_1 := mean(\cdot)$	$S_2 := Var(\cdot)$	$S_3 := Median(\cdot)$	$S_4 := \sqrt{Var(\cdot)}$
μ	5.631	0.001	0.0131	-0.001
σ^2	-0.002	3.835	0.008	0.0015

As we will use the projected summary statistics into ABC method with rejection process, which mean we will calculate the distance between the projected summary statistics of observation and simulated projected summary statistics, so the intercept of the linear regression term does not matter in such cases. We can have the value of the linear regression estimate of summary statistics corresponding to μ is $S_\mu = 5.645 \times S_1 + 6.303e^{-5} \times S_2$ in table 2.6. Similar way, we can have value of each summary statistic based on the projected summary statistics weights.

From the Table 2.6, we can find the projected summary statistics value of parameter μ depends mainly on the value of the mean with the regression coefficient for the variance contributing less than 0.00001 to μ; the opposite is the case for the parameter σ^2 which is mainly based on the value of the variance. This follows the theoretical expectation that the summary statistic, mean and variance, is sufficient for the parameter μ and σ^2 respectively. A similar situation holds in the case of the summary statistics based on the median and standard deviation. When we use all four summary statistics to construct projected values for parameters, as in Table 2.8, we can see that the projected summary statistics value of parameter μ is more likely to depend on the mean value rather than median; similarly, the projected summary statistics value of parameter σ^2mainly depends on the summary statistics variance. As all summary statistics are informative for the parameter set of $\theta := (\mu, \sigma^2)$, this example suggests that when a number of summary statistics are used, the semi-automatic regression method can obtain the estimated value of $\theta := (\mu, \sigma^2)$ with the most informative summary statistics. Through this method, we can decrease the number of summary statistics used in the ABC framework to the number of parameters that we are interested in by using projected summary statistics instead of the original summary statistics.

Continuing on, I present the result of the joint distribution of posterior ABC samples from the different methods in Figure 2.9. Here I also present the result from the standard rejection ABC method using sufficient summary statistics to compare with the semi-automatic regression method employing different sets of summary statistics and summarise the ABC posterior mean value of μ and σ^2 as estimates of μ and σ^2 in Table 2.9.

Table 2.9: The mean value of $\pi_{ABC}(\theta|\mathbf{y})$ by using different projection summary statistics

$\mathbb{E}(\pi(\theta\|\mathbf{y}))$	rejection ABC	$Proj\{S_1, S_2\}$	$Proj\{S_3, S_4\}$	$Proj\{S_1, S_2, S_3, S_4\}$
$\mu = 0.888$	0.754	0.887	0.584	0.887
$\sigma^2 = 4.760$	4.750	4.722	4.876	4.723

From Table 2.9 and Figure 2.9 we can see that compared with the standard rejection method, the semi-automatic regression method improves the performance of ABC significantly. Especially when using the sufficient summary statistics, the ABC posterior mean of (μ, σ^2) is very close to its theoretical posterior mean. Comparing with the rejection method using all summary statistics, the output from the semi-automatic regression method using all summary statistics is closer to its theoretical value than using those summary statistics directly into ABC process. However, the output from the semi-automatic regression method using median and standard deviation, which is informative but not sufficient summary statistics, is a biased result. Especially for inferring parameter μ, in Figure 2.9, the theoretical posterior mean is nearly not contained in the highest density curve of the posterior joint distribution of (μ, σ^2) and the ABC posterior mean (the blue point in Figure 2.9) is far away from its theoretical posterior mean (the red point in Figure 2.9). Besides, the result from using all informative summary statistics on the semi-automatic

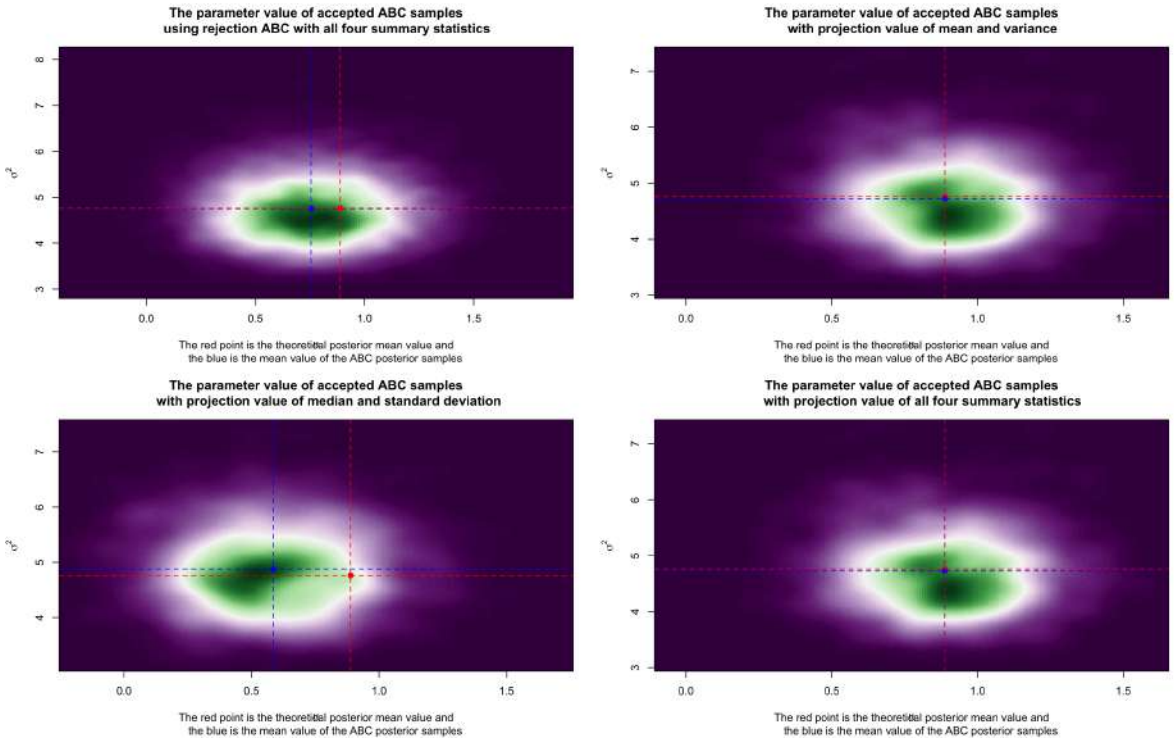

Figure 2.9: The comparison with different the set of summary statistics based semi-automatic regression method. The blue point in the figure is the ABC posterior mean of μ and σ^2 and the red points is the theoretical posterior mean of the μ and σ^2

regression method is quite similar to the result from the semi-automatic regression method only using sufficient summary statistics. The performance is less biased and more precise than the standard rejection method using sufficient summary statistics which are presented in the Table 2.4 and Table 2.5.

Two main conclusions may be drawn. Firstly, although the semi-automatic method improves the ABC performance significantly, if the summary statistics used in projection are not quite informative, it may also lead to a biased result. Secondly, the performance of the semi-automatic method is not much affected by increasing the number of summary statistics. Besides that, even using sufficient summary statistics, the projection summary statistics based on sufficient summary statistics also leads to better inferences than directly using sufficient summary statistics the standard rejection ABC method, at least in this specific numerical example. In particular, normally we are unable to know what are the sufficient or the most informative summary statistics for a complicated model, and the semi-automatic method encourages us to employ more informative summary statistics into the ABC framework and find the best set of summary statistics by a proper projection method. It empowers the summary statistics to capture more information, and at the same time, eliminates the effect from the curse of dimensionality.

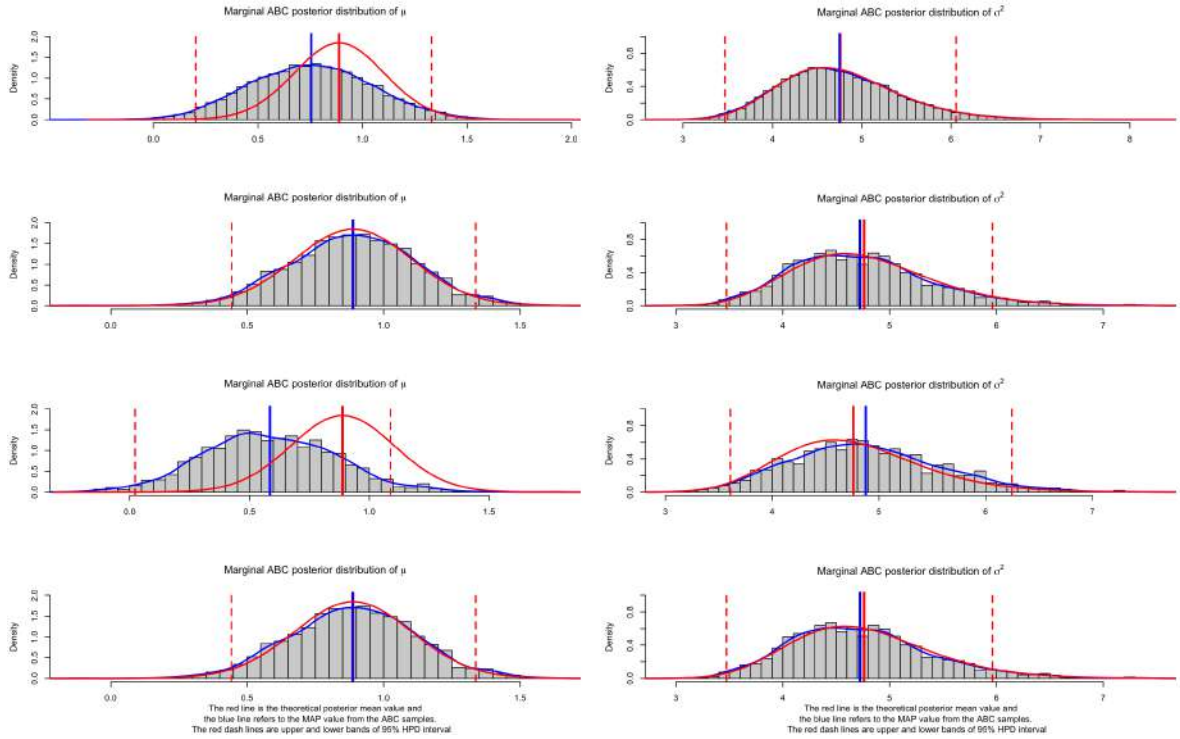

Figure 2.10: The marginal posterior distribution of μ and σ^2. The first row is the result from the rejection ABC method with summary statistics mean, variance, median and standard deviation. The second row is the result from the semi-automatic regression method with summary statistics mean and variance. The third row is the result from the semi-automatic regression method with summary statistics median and standard deviation. The bottom row is the result from the semi-automatic regression method with summary statistics mean, variance, median and standard deviation. The total number of simulation is 10^7 and $\epsilon = 0.1$. The blue vertical lines are the ABC posterior mean of μ and σ^2 from the accepted ABC samples and the red solid vertical lines are the theoretical posterior means of μ and σ^2. The red dotted vertical lines are the 95% HPD intervals of the theoretical posterior distributions of μ and σ^2. The blue curve is the smooth density curve from the accepted ABC samples of μ and σ^2 and the red curve is the theoretical marginal posterior density curve of μ and σ^2.

2.2.4 Regression-Adjustment Techniques

The regression-adjustment technique is a sort of post-processing technique. It was introduced by Beaumont et al. in 2002[15]. They suggested using local linear regression with weights from an Epanechnikov kernel to correct posterior samples. When given M samples from an initial approximation to the ABC posterior,

$$\{\theta_i, s(\mathbf{x}_i)\} \sim p_\epsilon(\theta, s(\mathbf{x})|s(\mathbf{y})) \tag{2.32}$$

where $i = 1, 2, \cdots, M$, we can use regression to obtain an estimate of the expected value of parameters given simulated summary statistics, which is denoted as $\hat{\mathbb{E}}(\theta|s(\mathbf{x}))$. Then we use this estimate to make an adjustment for each posterior sample as

$$\theta_i^* = \theta_i - \hat{\mathbb{E}}(\theta|s(\mathbf{x})) + \hat{\mathbb{E}}(\theta|s(\mathbf{y})) \tag{2.33}$$

Blum and François further developed this method and modified it by introducing a correction for heteroscedasticity which is from an additional regression. They used one more regression on the residual term to obtain an estimate of the standard deviation, which is $\hat{\sigma}(\theta|s(\mathbf{x}))$. Thus they enable an adjustment on parameters is,

$$\theta_i^* = \frac{\hat{\sigma}(\theta|s(\mathbf{y}))}{\hat{\sigma}(\theta|s(\mathbf{x}))}(\theta_i - \hat{\mathbb{E}}(\theta|s(\mathbf{x}))) + \hat{\mathbb{E}}(\theta|s(\mathbf{y})) \tag{2.34}$$

In Figure 2.11 and Figure 2.12, I present the different ABC methods to generate ABC posterior samples with the ABC posterior mean of parameters we are interested in. In all trials, the number of simulations is 10^7 and the smooth kernel used here is an Epanechnikov kernel. In contrast with the former section, here I choose the acceptance proportion of total simulation to be 0.001 instead of using tolerance $\epsilon = 0.1$ which is used in all the former Normal distribution examples. Using the acceptance proportion of simulated points can make sure the number of total accepted ABC samples is the same for different trials and makes the results from those post-adjustment methods more clear.

The outputs from standard rejection ABC with summary statistics using the acceptance proportion is 0.001 are similar to those former result from standard rejection ABC with summary statistics using tolerance $\epsilon = 0.1$. The marginal posterior distribution of σ^2 close to its theoretical posterior density but the histogram of the marginal posterior distribution μ presents a bias. Such a situation improves when using the projection summary statistics from the semi-automatic regression method instead of the former four summary statistics. The improvement of semi-automatic regression method I have discussed in the former section 2.2.3.2, and here we notice that the ABC posterior mean of μ is nearly as same as the theoretical marginal posterior mean of the parameter μ, but the shape of the marginal posterior distribution is broader than its theoretical marginal posterior density curve. The local linear regression method leads to a

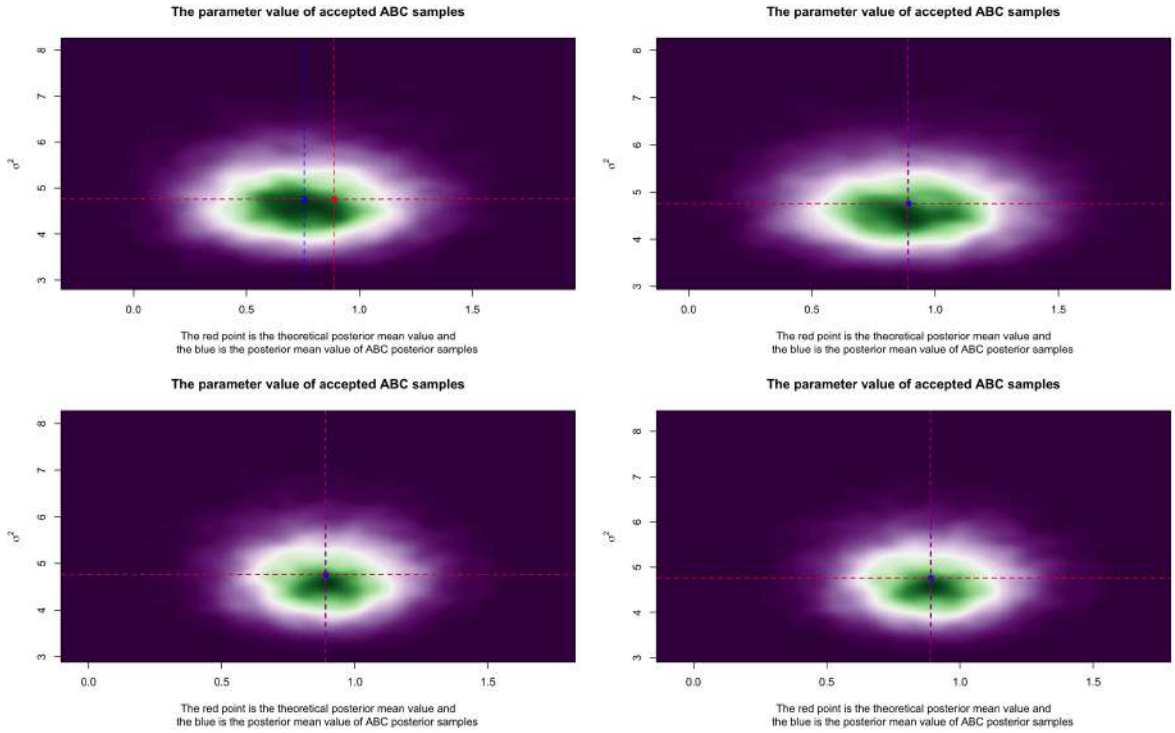

Figure 2.11: Comparison of the performance of different ABC method using total a simulation number of 10^7 with the acceptance proportion of total simulation to be 0.001. The top left is the result from standard rejection method. The top right is the result from standard rejection method using semi-automatic summary statistics. The bottom left gives the result from the local linear regression method using semi-automatic summary statistics without heteroscedasticity correction. The bottom right shows the result from the local linear regression method using semi-automatic summary statistics with heteroscedasticity correction. The blue point in the figure is the ABC posterior mean of μ and σ^2 and the red point is the theoretical posterior mean of the μ and σ^2

smaller variance for the posterior distribution. By using the local linear regression method with semi-automatic summary statistics, the ABC posterior marginal distribution is precise in comparison with its theoretical marginal posterior density curve. Regression adjustment with heteroscedasticity correction performs in a similar way in my trials, and there does not exist an obvious difference between the result from using regression adjustment with heteroscedasticity correction and the result from using local linear regression directly. The regression-adjustment technique is practical, useful and easily implemented. However, some publications have pointed out that when the observations are unlikely under the prior prediction distributions, it maybe give misleading results[117]. So there is a careful model choice needed before using such a post-processing technique.

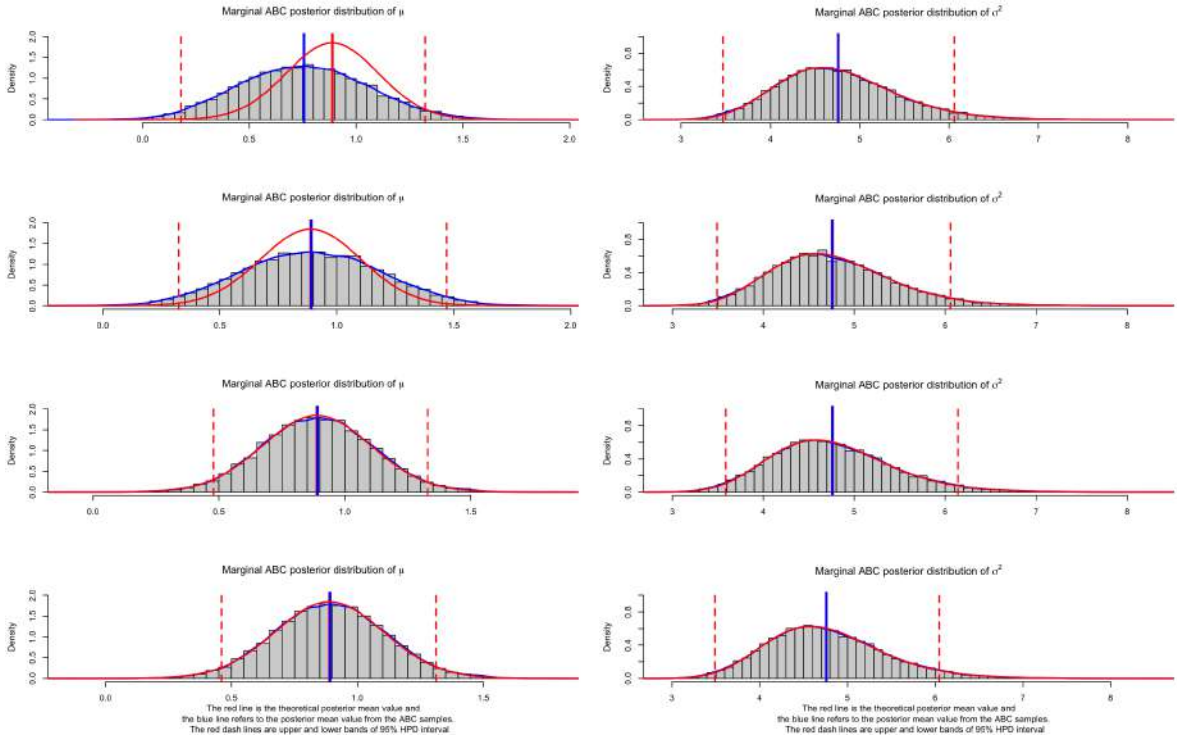

Figure 2.12: The marginal posterior distribution of different ABC method. The top row is the result from standard rejection method. The second row is the result from standard rejection method using semi-automatic summary statistics. The third row is the result from the local linear regression method using semi-automatic summary statistics without heteroscedasticity correction. The bottom row is result from the local linear regression method using semi-automatic summary statistics with heteroscedasticity correction. The blue vertical lines are the ABC posterior mean of μ and σ^2 from the accepted ABC samples and the red solid vertical lines are the theoretical posterior means of μ and σ^2. The red dotted vertical lines are the 95% HPD intervals of the theoretical posterior distributions of μ and σ^2. The blue curve is the smooth density curve from the accepted ABC samples of μ and σ^2 and the red curve is the theoretical marginal posterior density curve of μ and σ^2.

2.3 Summarize of the Chapter

We often face complex high dimensional models when dealing with the population problem, which challenges the classical statistical inference method. Especially, in some circumstance, the likelihood is intractable or computational infeasible. Thanks to the Monte Carlo method, we now have some approaches to handle this problem. In this chapter, I mainly introduced two important Monte Carlo based methods, i.e., MCMC and ABC. Both of them are widely used in the modern population genetics model-based parameter inference problem.

In the section of MCMC, I focus on introducing pseudo marginal Metropolis-Hastings method which is developed to solve the problem that the likelihood function is infeasible in calculating

acceptance ratio. It is a usual case, especially we employ a hidden Markov model to make the modelling of allele frequencies dynamic based on the Wright-Fisher model. In Chapter three and Chapter four, I will illustrate how to use the particle filter method to calculate Monte Carlo estimates of the likelihood function in pseudo marginal Metropolis-Hastings process and using this method to infer the selection coefficient and allele age based on different single-locus Wright-Fisher model and two-locus Wright-Fisher model.

Besides that, ABC is also widely used in population genetics problems, especially model choice and parameter inference. In the section of ABC, I mainly discussed the performance of different ABC techniques and the importance of the choice of summary statistics. In Chapter 5, I will illustrate an algorithm combining ABC with expectation propagation, which is a method intuitively generated from machine learning. By using the advanced ABC method, today we can effectively and efficiently make the model choice and infer parameter based on the complex demographic structure using a large dataset. I will discuss more detail in chapter five.

CHAPTER 3

BAYESIAN INFERENCE OF NATURAL SELECTION AND ALLELE AGE FROM ALLELE FREQUENCY TIME SERIES DATA

Ancient DNA(aDNA) preparation and sequencing techniques have made it more accessible for an increasing amount of high-quality time serial samples. Those time serial samples provide valuable information about the allele frequency trajectory, which allows us to have a better understanding of the evolutionary history as I have discussed in the former chapter. In this chapter, I will focus on work that has arisen from a collaboration with Zhangyi He and Feng Yu. My contribution towards this work is developing a two-step method combining co-estimating the initial underlying population frequency and selection coefficient using particle marginal Metropolis-Hastings and inferring the allele age by solving the corresponding Kolmogorov backward equation. A related investigation has been submitted for Genetics.

3.1 Introduction

One of the most important applications of ancient DNA (aDNA) is to study the action of natural selection because it enables us to directly track the change in allele frequencies over time, which is closely related to the strength of natural selection. Several studies over the past decade have been published capitalizing on the temporal aspect of aDNA data to characterize the process of natural selection, *e.g.*, Mathieson et al. [77] utilized aDNA data to identify candidate loci under natural selection in European humans. Malaspinas [73] provided an excellent review of existing methods to study natural selection using aDNA samples.

Several statistical methods have been developed to infer the action of natural selection from time-series data of allele frequencies obtained via aDNA. Initially, Bollback et al. [17] devised a likelihood-based approach to estimate the selection coefficient and the population size from

time-series data of allele frequencies assuming a Wright-Fisher model. In Bollback et al. [17], the population allele frequency was modelled as a latent variable in a hidden Markov model (HMM) framework, in which the sample allele frequency drawn from the underlying population at a given time point was treated as a noisy observation of the underlying population allele frequency. To incorporate natural selection, the Wright-Fisher model was approximated with a standard diffusion process, and the transition probabilities of the allele frequencies at each given time point were calculated by solving the Kolmogorov forward equation associated with the transition probability density function of the diffusion approximation through a finite difference method. Ludwig et al. [72] analysed the time series aDNA data associated with horse coat coloration with the method of Bollback et al. [17] and found that natural selection strongly acted on the locus encoding for the Agouti signalling peptide (ASIP) and the locus encoding for the melanocortin 1 receptor (MC1R).

Malaspinas et al. [74] extended the Bollback et al. [17] framework to jointly estimate the selection coefficient, the population size and the allele age from allele frequency time series data. Allele age is the time since the allele was created by mutation, which is an omnipresent parameter in population genetics and plays an important role in determining the sojourn time of a beneficial mutation along with the selection coefficient [see 106, for a detailed review]. The co-estimation of the allele age allows us to avoid the assumption on the latent population allele frequency at the first sampling time point in Bollback et al. [17], where it was assumed to be equal to the observed sample allele frequency or to be uniformly distributed. In Malaspinas et al. [74], the transition probabilities of the allele frequencies at each given time point were calculated by approximating the diffusion approximation of the Wright-Fisher model with a one-step Markov process. Steinrücken et al. [110] proposed an extension of the Bollback et al. [17] framework based on a spectral representation of the diffusion approximation of the Wright-Fisher model devised by Song and Steinrücken [108], which allows for a more general diploid model of natural selection such as the case of under- or overdominance. Besides, the method of Steinrücken et al. [110] enables us to avoid the discretisation of the state space in both Bollback et al. [17] and Malaspinas et al. [74], which is required to be fine enough to get a reliable approximation of the transition probabilities of the allele frequencies at each given time point and is strongly dependent on the underlying population genetic parameters. However, Steinrücken et al. [110] could not estimate the allele age since recurrent mutation was allowed in their model. Under a recurrent evolution model, the mutation repeatedly occurs and therefore the allele age has a very wide distribution.

In this work, I will introduce a novel Bayesian framework for the inference of both natural selection and allele age via the time series data of allele frequencies. This approach proceeds in two steps. First, we estimate the posterior probability distribution for the selection coefficient and the allele frequency of the underlying population at the first sampling time point of the non-zero observation with the particle marginal Metropolis-Hastings (PMMH) algorithm of

Andrieu et al. [3]. In the second step, by using the posterior from the first step, we obtain the Bayesian estimates of the selection coefficient and the allele age jointly. Unlike Schraiber et al. [99], our method generates the allele frequency trajectories of the underlying population from the Wright-Fisher diffusion directly, rather than the Wright-Fisher diffusion bridge. With the particle filter technique, trajectories that give rise to the population allele frequencies very different from the sample allele frequencies at the given time points are assigned very small weights and are therefore unlikely to be resampled. Intuitively, the results from our method are similar to Wright-Fisher diffusion, our method avoids the analytical and programming complications. Our approach can be highly efficient in the sense that we avoid exploring the unbounded state space of the allele age in our Bayesian procedure, and can be easily extended to incorporate variable population sizes, general models of natural selection and more complicated demographic histories such as inferred by Der Sarkissian et al. [26]. Our method can be highly efficient in the sense that no realisation we run from the first sampling time point of the non-zero observation backward in time is thrown away. Secondly, our model is parsimonious, in the sense that the only parameters we build into the model are parameters we try to estimate. There are no arbitrary or hidden parameters. Here we evaluate the performance of our Bayesian inference procedure with extensive simulations, showing that our method allows for the accurate inference of natural selection and allele age from time-series data of allele frequencies, and then employ it to analyse the time serial sample obtained via aDNA associated with horse coat coloration from Ludwig et al. [72]. Further, it can be readily extended to incorporate variable population size and a more general model of natural selection.

3.2 Wright-Fisher diffusion

3.2.1 Diffusion notations

In this section, we begin with a brief review of the Wright-Fisher diffusion for a single locus evolving under natural selection. Let us consider a panmictic population of randomly mating diploid individuals at a single autosomal locus $\mathscr{A}$ evolving under natural selection according to the one-locus Wright-Fisher model with selection [30], for which we assume discrete-time and nonoverlapping generations. At locus $\mathscr{A}$, there are two possible allele types, labeled $\mathscr{A}_1$ and $\mathscr{A}_2$. We attach the symbol $\mathscr{A}_1$ to the mutant allele, which is assumed to arise only once in the population (*i.e.*, there is no recurrent mutation) and be favored by natural selection, and we attach the symbol $\mathscr{A}_2$ to the ancestral allele, which is assumed to exist in the population originally.

Suppose that natural selection takes the form of viability selection, and the relative viabilities of the three possible genotypes at each locus, *e.g.*, genotypes $\mathscr{A}_1\mathscr{A}_1$, $\mathscr{A}_1\mathscr{A}_2$ and $\mathscr{A}_2\mathscr{A}_2$ at a given locus $\mathscr{A}$, are taken to be 1, $1-h_{\mathscr{A}}s_{\mathscr{A}}$ and $1-s_{\mathscr{A}}$, respectively, where $s_{\mathscr{A}} \in [0,1]$ is the selection coefficient (*i.e.*, directional selection) and $h_{\mathscr{A}} \in [0,1]$ is the dominance parameter. We designate the population size by N, which is assumed to be fixed.

3.2.2 Wright-Fisher diffusion with selection

We consider a scaling limit of the Wright-Fisher model where the population size N goes to infinity while the unit of time is rescaled by $2N$ and the rescaled selection coefficient $\alpha_{\mathscr{A}} = 2Ns_{\mathscr{A}}$ is kept constant. According to Durrett [30], the allele frequency trajectory through time follows a standard diffusion approximation of the one-locus Wright-Fisher model with selection as the population size N goes to infinity, called the one-locus Wright-Fisher diffusion with selection. Typically, the Wright-Fisher diffusion is formulated in terms of the partial differential equation (PDE) satisfied by its transition probability density function [*e.g.*, 17, 47, 110], but in this work we characterise it as the solution of the stochastic differential equation (SDE) instead [*e.g.*, 99]. More specifically, we let $X(t) \in [0,1]$ denote the frequency of the mutant allele in the population at time t, which satisfies the following SDE

$$dX(t) = \mu(X(t))dt + \sigma(X(t))dW(t), \qquad t \geq t_{\mathscr{A}} \tag{3.1}$$

where the drift coefficient $\mu(x)$ is

$$\mu(x) = \alpha_{\mathscr{A}} x(1-x)((1-h_{\mathscr{A}}) - (1-2h_{\mathscr{A}})x), \tag{3.2}$$

the diffusion coefficient $\sigma(x)$ is

$$\sigma(x) = \sqrt{x(1-x)}, \tag{3.3}$$

and $W(t)$ is a standard Brownian motion.

To specify the Wright-Fisher diffusion evolving under natural selection with the additional population genetic quantity of interest, the allele age, we need to specify certain conditions at time $t_{\mathscr{A}}$, the time when the mutant allele $\mathscr{A}_1$ arises in the population here. Schraiber et al. [99] took the initial mutant allele frequency $X(t_{\mathscr{A}})$ to be some small but arbitrary value $x_{\mathscr{A}}$, which was found to be feasible in their method but it is nevertheless slightly unsatisfying. In our Bayesian inference procedure, we take $X(t) = 0$ for $t < t_{\mathscr{A}}^{-} = 0$ and assume that a new mutation arises at locus $\mathscr{A}$ in a single individual in a population of N individuals at time $t_{\mathscr{A}}$, giving rise to the $\mathscr{A}_1$ allele, thereby we take $X(t_{\mathscr{A}}) = 1/(2N)$. In such a setup, there is no need to specify an arbitrary initial mutate allele frequency.

3.2.3 Euler-Maruyama scheme

Now we consider the problem of how to solve the SDE in Eqs. (3.1)-(3.3) numerically. A number of numerical approaches for SDEs have already been developed [see 64, for an excellent introduction], and the numerical method we adopt here is the commonly used Euler-Maruyama scheme, one of the most popular numerical methods for SDEs in practice due to its high efficiency and low complexity.

One of the simplest numerical approximations for the SDE is the Euler-Maruyama method. If we truncate Ito's formula of the stochastic Taylor series after the first order terms, we obtain the Euler method or Euler-Maruyama method as follows

We introduce a partition of the time interval $[t_{\mathscr{A}}, T]$, defined as

$$\Delta^{(M)}_{[t_{\mathscr{A}},T]} = \left\{ \tau_m : \tau_m = t_{\mathscr{A}} + \frac{T - t_{\mathscr{A}}}{M} m \text{ for } m = 0, 1, \ldots, M \right\}, \tag{3.4}$$

and then the Euler-Maruyama scheme can be formulated as

$$\hat{X}(\tau_m) = \hat{X}(\tau_{m-1}) + \mu(\hat{X}(\tau_{m-1}))\Delta\tau_{m-1} + \sigma(\hat{X}(\tau_{m-1}))\Delta W(\tau_{m-1}), \tag{3.5}$$

where $\Delta\tau_{m-1} = \tau_m - \tau_{m-1}$ and $\Delta W(\tau_{m-1}) = W(\tau_m) - W(\tau_{m-1})$ are independent and normally distributed with mean 0 and variance $\Delta\tau_{m-1}$. Notice that we need to incorporate the jump of size $1/(2N)$ in the simulated allele frequency trajectory at time $t_{\mathscr{A}}$, by taking the initial condition $\hat{X}(\tau_0) = 1/(2N)$.

The intuitive idea of Euler-Maruyama scheme to make computational simulation of SDEs is similar to Euler's method for ODEs. Euler's method is to approximate the solution using some fixed small constant Δt and then we can have $x(t + \Delta t) = x(t) + f(x,t)\Delta t$. Euler-Maruyama approximation converges with strong order 0.5 under Lipschitz and bounded growth conditions on the coefficients μ and σ, which were shown in Gikhman and Skorokhod [44]. Mil'shtein [83] showed that an Euler- Maruyama approximation of an Ito process converges with weak order 1.0 under conditions of sufficient smoothness. It is clear that weak order of convergence is greater than strong order of convergence in the Euler-Maruyama method [9].

3.3 Bayesian inference of natural selection and allele age

We demonstrate our Bayesian method for the inference of natural selection and allele age from time-series data of allele frequencies, including how to set up the HMM framework and how to compute the posterior probability distribution for the selection coefficient, the population size and the allele age with Monte Carlo techniques. Let us consider any single locus along the chromosome, labeled $\mathscr{A}$, and recall that the $\mathscr{A}_1$ allele is the mutant allele favoured by natural selection. The population genetic parameters of interest here are the selection coefficient $s_{\mathscr{A}}$, the dominance parameter $h_{\mathscr{A}}$, the population size N and the allele age $t_{\mathscr{A}}$ measured in the units of $2N$ generations, denoted by $\boldsymbol{\vartheta} = (s_{\mathscr{A}}, h_{\mathscr{A}}, t_{\mathscr{A}}, N)$. In this section, we introduce our HMM framework incorporated with the one-locus Wright-Fisher diffusion with selection and describe our Bayesian inference approach for estimating the parameters $\boldsymbol{\vartheta}$ from the time series data of allele frequencies.

3.3.1 Hidden Markov model

We employ an HMM model similar to that proposed in Bollback et al. [17]. We assume the underlying population evolves according to the Wright-Fisher diffusion in Eqs. (3.1)-(3.3) and

we model the observations as independent sampling from the underlying population at each given time point. Suppose that the available data here are always sampled from the underlying population at a finite number of distinct time points, at say $t_1 < t_2 < \ldots < t_K$, where the time is measured in units of $2N$ generations to be consistent with the time scale in the Wright-Fisher diffusion. At the k-th sampling time point, there are c_k mutant alleles at locus $\mathscr{A}$ found in the sample of n_k individuals drawn from the underlying population.

To achieve the estimates of the population genetic quantities of interest, we need to compute the posterior probability density function $p(\boldsymbol{\vartheta} \mid \boldsymbol{c}_{1:K}, \boldsymbol{n}_{1:K}, \boldsymbol{t}_{1:K})$, which can be obtained by conditioning and integrating over all possible allele frequency trajectories of the underlying population at each sampling time point [see, *e.g.*, 17, 74, 99, 110]. We let $x_{\mathscr{A}}$ and $\boldsymbol{x}_{1:K} = (x_1, x_2, \ldots, x_K)$ denote the allele frequency of the underlying population at the times $t_{\mathscr{A}}$ and $\boldsymbol{t}_{1:K}$, respectively. Note that we take $x_{\mathscr{A}} = 1/(2N)$ under the Wright-Fisher diffusion as we have stated in the previous section. In our method, we achieve the estimates of the parameters $\boldsymbol{\vartheta}$ from the posterior probability density function $p(\boldsymbol{\vartheta}, \boldsymbol{x}_{1:K} \mid \boldsymbol{c}_{1:K}, \boldsymbol{n}_{1:K}, \boldsymbol{t}_{1:K})$ rather than the posterior probability density function $p(\boldsymbol{\vartheta} \mid \boldsymbol{c}_{1:K}, \boldsymbol{n}_{1:K}, \boldsymbol{t}_{1:K})$. This procedure avoids integrating over all possible allele frequency trajectories of the underlying population at each sampling time point, which makes the computation far more efficient.

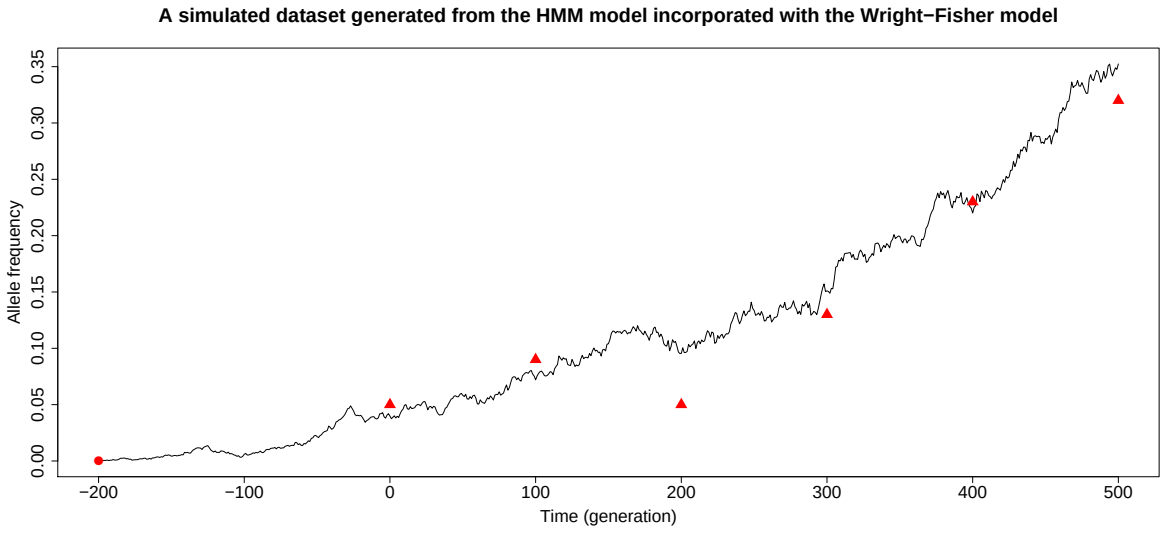

Figure 3.1: An example of the simulated dataset. We assume that the mutant allele $\mathscr{A}_1$ arises at frequency 0.0001 in the underlying population in generation $k_{\mathscr{A}} = -500$ (red filled circle) and simulate the mutant allele frequency trajectory of the underlying population using the one-locus Wright-Fisher model with selection (black line). From generation 0 to 500, we select 40 individuals from the underlying population every 100 generations (red filled triangle). In this illustration, we take $N = 5000$, $s_{\mathscr{A}} = 0.01$ and $h_{\mathscr{A}} = 0.5$.

Let t_{k^*} be the first time point amongst the sampling time points $\boldsymbol{t}_{1:K}$ at which the mutant allele $\mathscr{A}_1$ has been found in the sample. This means that $\boldsymbol{c}_{1:k^*-1}$ are all zero where $t_{\mathscr{A}} < t_{k^*}$, but $\boldsymbol{x}_{1:k^*-1}$ may not be all zero. So the joint posterior probability density function for the population genetic quantities of interest and the allele frequency trajectory of the underlying population can be written as

$$
\begin{aligned}
(3.6)\quad p(s_{\mathscr{A}}, h_{\mathscr{A}}, t_{\mathscr{A}}, \boldsymbol{x}_{1:K} \mid \boldsymbol{c}_{1:K}) &\propto p(s_{\mathscr{A}}, h_{\mathscr{A}}, t_{\mathscr{A}}, \boldsymbol{x}_{1:K}, \boldsymbol{c}_{1:K}) \\
&= p(\boldsymbol{c}_{1:K} \mid s_{\mathscr{A}}, h_{\mathscr{A}}, t_{\mathscr{A}}, \boldsymbol{x}_{1:K}) p(\boldsymbol{x}_{1:K} \mid x_{k^*}, s_{\mathscr{A}}, h_{\mathscr{A}}, t_{\mathscr{A}}) p(x_{k^*}, s_{\mathscr{A}}, h_{\mathscr{A}}, t_{\mathscr{A}}) \\
&= p(\boldsymbol{c}_{1:K} \mid \boldsymbol{x}_{1:K}) p(\boldsymbol{x}_{1:k^*} \mid s_{\mathscr{A}}, h_{\mathscr{A}}, t_{\mathscr{A}}) p(\boldsymbol{x}_{k^*+1:K} \mid x_{k^*}, s_{\mathscr{A}}, h_{\mathscr{A}}, t_{\mathscr{A}}) p(s_{\mathscr{A}}, h_{\mathscr{A}}, t_{\mathscr{A}}) \\
&= p(\boldsymbol{c}_{1:k^*-1} = 0 \mid \boldsymbol{x}_{1:k^*-1}) p(\boldsymbol{x}_{1:k^*} \mid s_{\mathscr{A}}, h_{\mathscr{A}}, t_{\mathscr{A}}) p(\boldsymbol{c}_{k^*:K} \mid \boldsymbol{x}_{k^*:K}) p(\boldsymbol{x}_{k^*+1:K} \mid x_{k^*}, s_{\mathscr{A}}, h_{\mathscr{A}}) p(s_{\mathscr{A}}, h_{\mathscr{A}}, t_{\mathscr{A}}) \\
&= p_1(\boldsymbol{x}_{k^*:K}, s_{\mathscr{A}}, h_{\mathscr{A}}) p_2(\boldsymbol{x}_{1:k^*}, t_{\mathscr{A}}),
\end{aligned}
$$

where we define

$$
\begin{aligned}
p_1(\boldsymbol{x}_{k^*:K}, s_{\mathscr{A}}, h_{\mathscr{A}}) &= p(\boldsymbol{c}_{k^*:K} \mid \boldsymbol{x}_{k^*:K}) p(\boldsymbol{x}_{k^*+1:K} \mid x_{k^*}, s_{\mathscr{A}}, h_{\mathscr{A}}) p(s_{\mathscr{A}}, h_{\mathscr{A}}) p(x_{k^*}) \\
p_2(\boldsymbol{x}_{1:k^*}, t_{\mathscr{A}}) &= \frac{1}{p(x_{k^*})} p(\boldsymbol{c}_{1:k^*-1} = 0 \mid \boldsymbol{x}_{1:k^*-1}) p(\boldsymbol{x}_{1:k^*} \mid s_{\mathscr{A}}, h_{\mathscr{A}}, t_{\mathscr{A}}) p(t_{\mathscr{A}} \mid s_{\mathscr{A}}, h_{\mathscr{A}}),
\end{aligned}
$$

and $p(x_{k^*})$ is an arbitrary prior distribution on the hidden parameter x_{k^*}, which can be taken to be uniform on $[0,1]$.

Our Bayesian inference procedure for the estimation of the parameters $(s_{\mathscr{A}}, h_{\mathscr{A}}, t_{\mathscr{A}})$ from the time series data of allele frequencies can therefore proceed in two steps: the first step is to employ the PMMH algorithm to estimate $p_1(\boldsymbol{x}_{k^*:K}, s_{\mathscr{A}}, h_{\mathscr{A}})$ of Eq. (3.6), and the second step is to combine this estimate with $p_2(\boldsymbol{x}_{1:k^*}, t_{\mathscr{A}})$ to achieve the MAP estimates of the parameters $(s_{\mathscr{A}}, h_{\mathscr{A}}, t_{\mathscr{A}})$.

3.3.2 Particle marginal Metropolis-Hastings

For the term $p_1(\boldsymbol{x}_{k^*:K}, s_{\mathscr{A}}, h_{\mathscr{A}})$ in Eq. (3.6), $p(s_{\mathscr{A}}, h_{\mathscr{A}})$ and $p(x_{k^*})$ are the prior probability density functions for the parameters $(s_{\mathscr{A}}, h_{\mathscr{A}})$ and x_{k^*}, respectively. They can be taken to be uniform if the prior knowledge of these parameters is poor. The term $p(\boldsymbol{x}_{k^*+1:K} \mid x_{k^*}, s_{\mathscr{A}}, h_{\mathscr{A}})$ is the probability density function for the population allele frequency trajectory at the times $\boldsymbol{t}_{k^*+1:K}$, and $p(\boldsymbol{c}_{k^*:K} \mid \boldsymbol{x}_{k^*:K})$ is the probability density function for the observations conditional on the population allele frequency trajectory at the times $\boldsymbol{t}_{k^*:K}$. Since the Wright-Fisher diffusion is a Markov process, we have

$$
(3.7)\qquad p(\boldsymbol{x}_{k^*+1:K} \mid x_{k^*}, s_{\mathscr{A}}, h_{\mathscr{A}}) = \prod_{k=k^*}^{K-1} p(x_{k+1} \mid x_k; s_{\mathscr{A}}, h_{\mathscr{A}}),
$$

where $p(x_{k+1} \mid x_k; s_{\mathscr{A}}, h_{\mathscr{A}})$ is the transition probability density function of the Wright-Fisher diffusion between two consecutive sampling time points for $k = k^*, k^*+1, \ldots, K-1$. Given the allele frequency trajectory of the underlying population, the observations at each sampling time

point are independent of one another and follow the binomial distribution, which implies that

$$p(\boldsymbol{c}_{k^*:K} \mid \boldsymbol{x}_{k^*:K}) = \prod_{k=k^*}^{K} p(c_k \mid x_k), \tag{3.8}$$

where

$$p(c_k \mid x_k) = \frac{(2n_k)!}{c_k!(2n_k - c_k)!} x_k^{c_k} (1-x_k)^{2n_k - c_k} \tag{3.9}$$

for $k = k^*, k^*+1, \ldots, K$.

In order to estimate the probability density function $p_1(\boldsymbol{x}_{k^*:K}, s_{\mathscr{A}}, h_{\mathscr{A}})$, we follow the PMMH algorithm of Andrieu et al. [3]. More specifically, we first draw a sample of the parameters $(x_{k^*}, s_{\mathscr{A}}, h_{\mathscr{A}})$ from the prior $p(s_{\mathscr{A}}, h_{\mathscr{A}})p(x_{k^*})$, and then we run a bootstrap particle filter with initial frequency x_{k^*} at time t_{k^*} and selection coefficients $(s_{\mathscr{A}}, h_{\mathscr{A}})$.

The bootstrap particle filter is an iterative method for carrying out Bayesian inference for hidden Markov models. For example, there is an unobserved Markov process $x_0, x_1, \ldots, x_T$ governed by a transition kernel $p(x_{t+1}|x_t)$ is partially observed via some measurement model $p(y_t|x_t)$ leading to observed data $y_1, y_2, \ldots, y_T$. The intuitive idea of the bootstrap particle filter is to make inference for the hidden states $x_{0:T}$ given the data $y_{1:T}$. The method is a simple application of the importance resampling technique. At each time t, we assume that we have an approximating sample from $p(x_t|y_{1:t})$ and use importance resampling to generate an approximate sample from $p(x_{t+1}|y_{1:t+1})$ [3].

Then make a jump on $(x_{k^*}, s_{\mathscr{A}}, h_{\mathscr{A}})$ and repeat the following steps until a sufficient number of samples have been obtained:

Step 1: Draw a candidate sample of the parameters $(x^*_{k^*}, s^*_{\mathscr{A}}, h^*_{\mathscr{A}})$ from the proposal $q(x^*_{k^*}, s^*_{\mathscr{A}}, h^*_{\mathscr{A}} \mid x_{k^*}, s_{\mathscr{A}}, h_{\mathscr{A}})$. The proposal distribution $q(x^*_{k^*}, s^*_{\mathscr{A}}, h^*_{\mathscr{A}} \mid x_{k^*}, s_{\mathscr{A}}, h_{\mathscr{A}})$ can be taken to be a truncated random walk on the parameter space $(x_{k^*}, s_{\mathscr{A}}, h_{\mathscr{A}})$.

Step 2: Run a bootstrap particle filter with parameters $(x^*_{k^*}, s^*_{\mathscr{A}}, h^*_{\mathscr{A}})$.

Step 3: Accept the candidate sample of the parameters $(x^*_{k^*}, s^*_{\mathscr{A}}, h^*_{\mathscr{A}})$ with the Metropolis-Hastings ratio

(3.10)

$$\alpha\left(x^*_{k^*}, s^*_{\mathscr{A}}, h^*_{\mathscr{A}} \mid x_{k^*}, s_{\mathscr{A}}, h_{\mathscr{A}}\right) = \frac{\hat{p}\left(\boldsymbol{c}_{k^*:K} \mid x^*_{k^*}, s^*_{\mathscr{A}}, h^*_{\mathscr{A}}\right) p\left(s^*_{\mathscr{A}}, h^*_{\mathscr{A}}\right) p(x^*_{k^*}) q\left(x_{k^*}, s_{\mathscr{A}}, h_{\mathscr{A}} \mid x^*_{k^*}, s^*_{\mathscr{A}}, h^*_{\mathscr{A}}\right)}{\hat{p}\left(\boldsymbol{c}_{k^*:K} \mid x_{k^*}, s_{\mathscr{A}}, h_{\mathscr{A}}\right) p\left(s_{\mathscr{A}}, h_{\mathscr{A}}\right) p(x_{k^*}) q\left(x^*_{k^*}, s^*_{\mathscr{A}}, h^*_{\mathscr{A}} \mid x_{k^*}, s_{\mathscr{A}}, h_{\mathscr{A}}\right)},$$

where $\hat{p}(\boldsymbol{c}_{k^*:K} \mid x_{k^*}, s_{\mathscr{A}}, h_{\mathscr{A}})$ is the particle filter's unbiased estimate of the marginal likelihood $p(\boldsymbol{c}_{k^*:K} \mid x_{k^*}, s_{\mathscr{A}}, h_{\mathscr{A}})$ and is equal to the product of average weights of all particles at the times $\boldsymbol{t}_{k^*:K}$.

By using PMMH illustrated above, we can generated a sufficient number of samples from posterior jointly distribution of parameters $(x_{k^*}, s_{\mathscr{A}}, h_{\mathscr{A}})$ and we choose 10^5 sets of parameters to

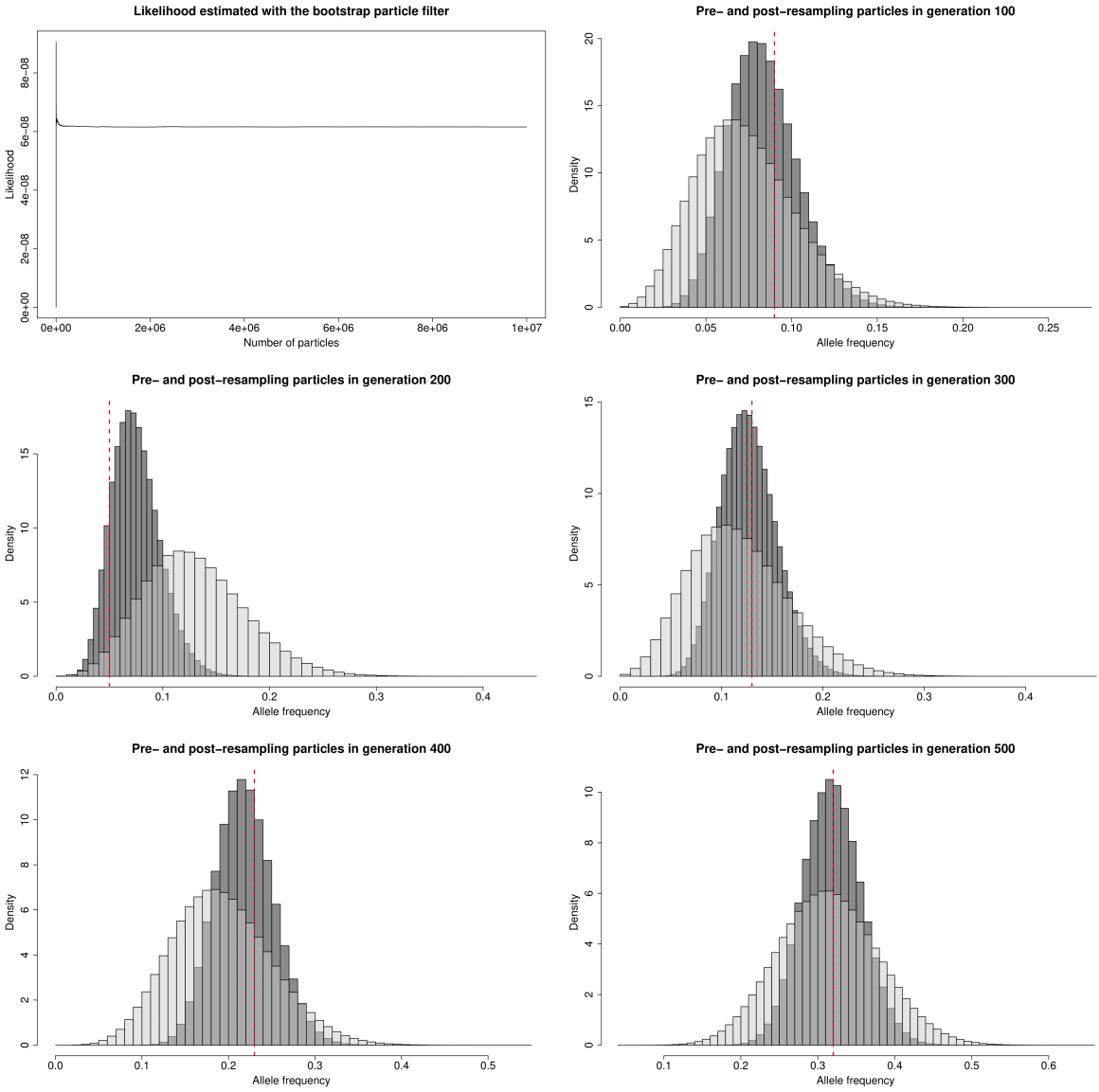

Figure 3.2: A PMMH resampling process example based on the simulated dataset presented in Figure 3.1. The light grey is the pre-resampling hidden state distribution. The dark grey is the post-resampling hidden state distribution. The red line is the observation allele frequency.

draw the density kernel in Figure 3.3. The trace plot of this PMMH process is shown in Figure 3.4

As I presented above, the Markov chain of parameters $(x_{k^*}, s_{\mathscr{A}}, h_{\mathscr{A}})$ mixed well and we can have a good estimate of the probability density function $p_1(\boldsymbol{x}_{k^*:K}, s_{\mathscr{A}}, h_{\mathscr{A}})$ by using a large number of samples from jointly distribution of parameters $(x_{k^*}, s_{\mathscr{A}}, h_{\mathscr{A}})$. For simplicity, this example is based on the simulated data set which I set the $h_{\mathscr{A}} = 0.5$ as I illustrated in Figure 3.1.

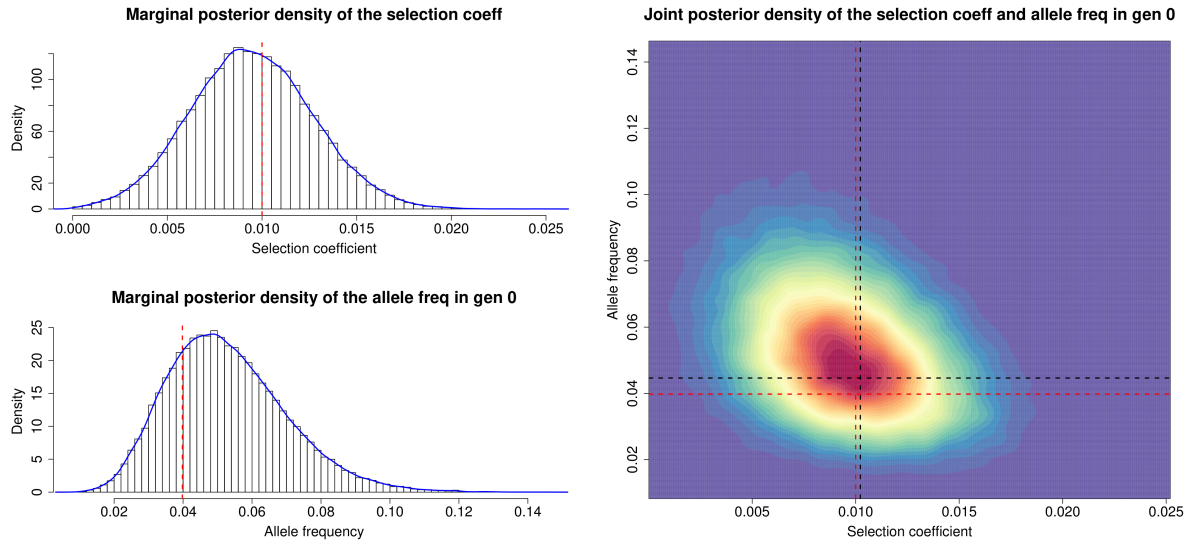

Figure 3.3: PMMH estimates of selection and initial frequency based on the simulated dataset presented in Figure 3.1. The red line dash line is the true value of selection coefficient $s_{\mathscr{A}}$ and initial population(underlying trajectory) allele frequency x_{k^*}. The black dash line is the MAP estimates of selection coefficient $\hat{s_{\mathscr{A}}}$ and initial population(underlying trajectory) allele frequency x_{k^*}

3.3.3 Backward Equation

Now we deal with the term $p_2(\boldsymbol{x}_{1:k^*}, t_{\mathscr{A}})$ in Eq. (3.6), given by

$$p_2(\boldsymbol{x}_{1:k^*}, t_{\mathscr{A}}) = \frac{1}{p(x_{k^*})} p(\boldsymbol{c}_{1:k^*-1} = 0 \mid \boldsymbol{x}_{1:k^*-1}) p(t_{\mathscr{A}} \mid s_{\mathscr{A}}, h_{\mathscr{A}}) p(\boldsymbol{x}_{1:k^*} \mid s_{\mathscr{A}}, h_{\mathscr{A}}, t_{\mathscr{A}})$$

The last term in the above can be expressed as the solution of the Kolmogorov Backward Equation. More precisely, let $t_0 = -\infty$ and $t_{\mathscr{A}} \in [t_{k'-1}, t_{k'})$ with $k' \leq k^*$, then

$$p(\boldsymbol{x}_{1:k^*} \mid s_{\mathscr{A}}, h_{\mathscr{A}}, t_{\mathscr{A}}) = p(x_{k'} \mid s_{\mathscr{A}}, h_{\mathscr{A}}, t_{\mathscr{A}}) \prod_{k=k'}^{k^*-1} p(x_{k+1} \mid x_k; s_{\mathscr{A}}, h_{\mathscr{A}}).$$

Let X be a diffusion process with parameters $(s_{\mathscr{A}}, h_{\mathscr{A}})$, with

$$dX(t) = \mu(X(t))dt + \sigma(X(t))dW(t), t \geq t_{\mathscr{A}},$$

then the term in the product above can be written as

$$p(x_{k+1} \mid x_k; s_{\mathscr{A}}, h_{\mathscr{A}}) = \lim_{\delta x \to 0} \frac{1}{\delta x} \mathbb{P}(X(t_{k+1}) \in (x_{k+1}, x_{k+1} + \delta x) \mid X(t_k) = x_k).$$

We can obtain this value by numerically solving the corresponding Kolmogorov backward equation:

$$-\frac{\partial}{\partial t} u(x,t) = \mu(x) \frac{\partial}{\partial x} u(x,t) + \frac{1}{2} \sigma(x)^2 \frac{\partial^2}{\partial x^2} u(x,t), \qquad t < t_{k+1}$$

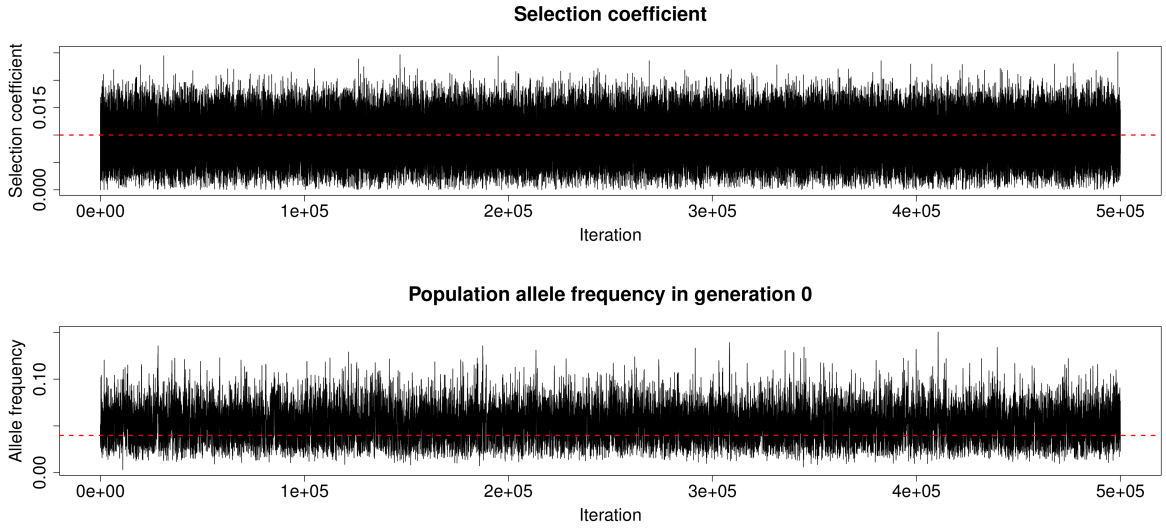

Figure 3.4: PMMH trace plot based on the simulated dataset presented in Figure 3.1. The red line dash line is the true value of selection coefficient $s_{\mathscr{A}}$ and initial population(underlying trajectory) allele frequency x_{k^*}

subject to the final condition $u(x,t_{k+1}) = \delta_{x_{k+1}}(x)$, where δ is the Dirac delta. The value $p(x_{k+1} \mid x_k; s_{\mathscr{A}}, h_{\mathscr{A}})$ is simply $u(x_k,t_k)$. The term outside the product can be written as

$$p(x_{k'} \mid s_{\mathscr{A}}, h_{\mathscr{A}}, t_{\mathscr{A}}) := \lim_{\delta x \to 0} \frac{1}{\delta x} \mathbb{P}\left(X(t_{k'}) \in (x_{k'}, x_{k'} + \delta x) \mid X(t_{\mathscr{A}}) = \frac{1}{2N} \right),$$

which can also be obtained by numerically solving the Kolmogorov backward equation, just like the terms in the product. So by using the KBE method, for each sample of parameter set $(x_{k^*}, s_{\mathscr{A}}, h_{\mathscr{A}})$ we can figure out a numerical solution for $X(t_0 = 0)$, here we also choose 10^5 samples to draw a joint density kernel of $(s_{\mathscr{A}}, t_{\mathscr{A}})$.

As I illustrated above, by using this two-step method which combines PMMH with KBE, we can have joint estimates of selection coefficient $s_{\mathscr{A}}$ and allele age $t_{\mathscr{A}}$ based on the simulated time series allele frequency dataset presented in Figure 3.1.

3.4 Simulation study

To evaluate the performance of our Bayesian inference approach, we run a few forward-in-time simulations of the one-locus Wright-Fisher model with selection [see, *e.g.*, 30] and assess the performance of our method by looking at the bias of our estimates and the root mean square error (RMSE). Both bias and the RMSE are statistics reflecting how estimates are biased from the expected values or the true values, where

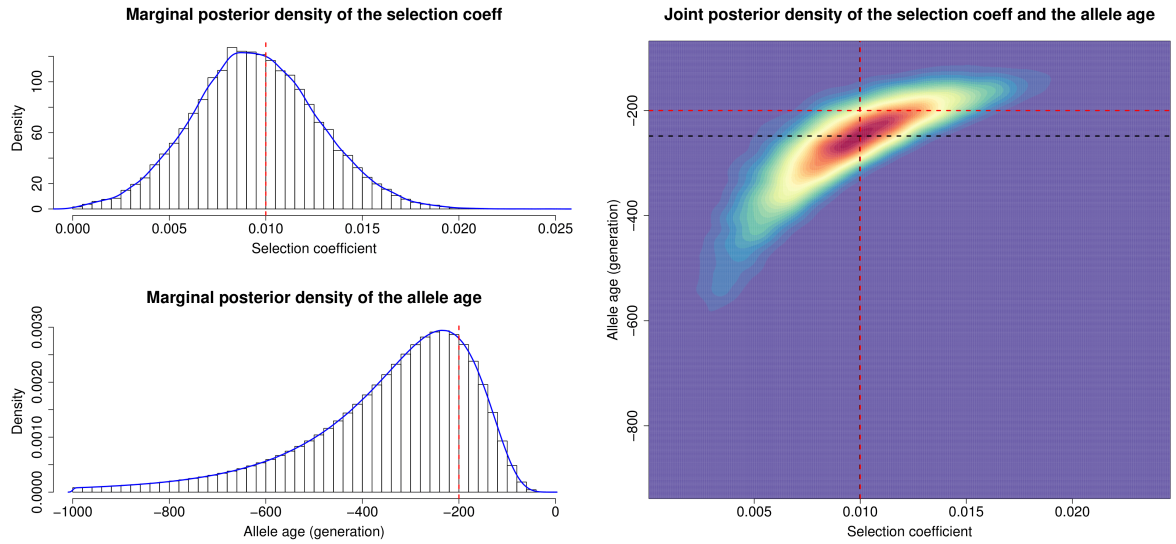

Figure 3.5: KBE output based on the simulated dataset presented in Figure 3.1. The red line dash line is the true value of selection coefficient $s_{\mathscr{A}}$ and allele age $t_{\mathscr{A}}$. The black dash line is the MAP estimates of selection coefficient $s_{\mathscr{A}}$ and allele age $t_{\mathscr{A}}$

$$RMSE = \sqrt{\frac{\sum_{i=1}^{N}\left(\hat{\theta}_i - \theta\right)^2}{N}}$$

$$\mathbf{E}(bias) = \frac{\sum_{i=1}^{N}\left(\hat{\theta}_i - \theta\right)}{N}$$

In what follows, unless otherwise noted, we pick a population size of $N = 5000$, set the allele age to $t_{\mathscr{A}} = -200$ measured in the units of generations, and fix the dominance parameters to be $h_{\mathscr{A}} = 0.5$ (*i.e.*, the heterozygous fitness is the arithmetic average of the homozygous fitness, called genic selection). In principle, however, the conclusions hold for any other values of the population size $N \in \mathbb{N}$, the allele age $t_{\mathscr{A}} \in \mathbb{Z}$ and the dominance parameter $h_{\mathscr{A}} \in [0,1]$.

For every simulated dataset, we use the one-locus Wright-Fisher model with selection to simulate the latent mutant allele frequency trajectory of the underlying population with the given values of the population genetic parameters $\boldsymbol{\vartheta} = (s_{\mathscr{A}}, h_{\mathscr{A}}, t_{\mathscr{A}}, N)$. After obtaining the simulated mutant allele frequency trajectory of the underlying population, we draw the observed mutant allele count independently at each sampling time point according to the binomial distribution in Equation 3.9. Here the observations are set to be taken every 100 generations from generation 0 to 500 each with 40 individuals. We consider the selection coefficients $s_{\mathscr{A}} \in \{0, 0.001, 0.005, 0.01\}$ and generate at least 200 simulated datasets for each set of the values of the population genetic parameters $\boldsymbol{\vartheta}$. Based on the results, I use a histogram figure and two box-plot for each parameter

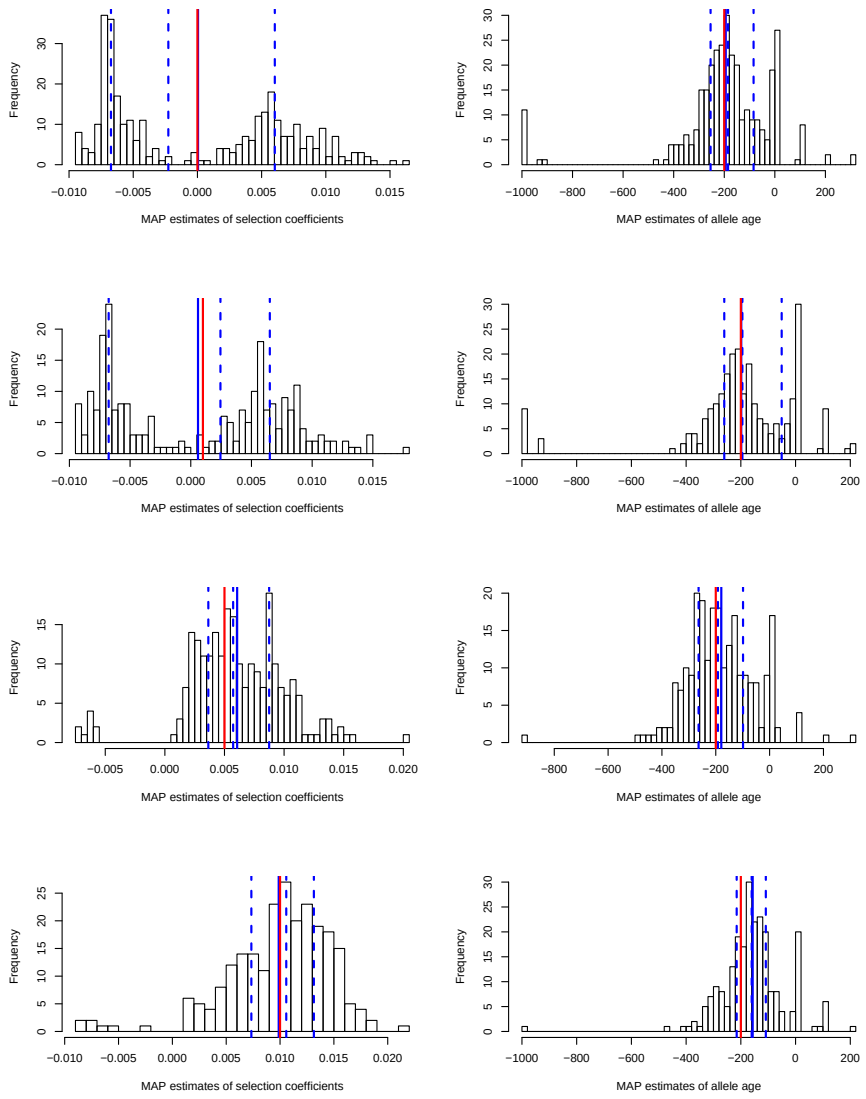

Figure 3.6: Histogram of the MAP estimates of the parameters selection coefficient and the allele age, i.e., $\left(s_{\mathscr{A}} \times 10^{-3}, t_{\mathscr{A}}\right)$. From the top to the bottom are different trials with true parameter values set are (0, -200), (1, -200), (5, -200), (10, -200), respectively. The three blue dash vertical lines are first quartile, median and third quartile values respectively. The solid blue line is the mean value of the MAP estimates and the red solid line is the true value for the parameter.

to present our method performance and summary the output in Table 3.1 and Table 3.2. I use maximum a posterior value as estimates of the parameter, in Table 3.1 and Table 3.2, I summarise the performance of estimates by using calculate the mean MAP estimates value along with the total number of replicates, which is in the third column and using the mean MAP estimates value to calculate the bias term, which is the discrepancy between the mean MAP estimates the value and the true parameter value. I also calculate RMSE based on those MAP estimates value and

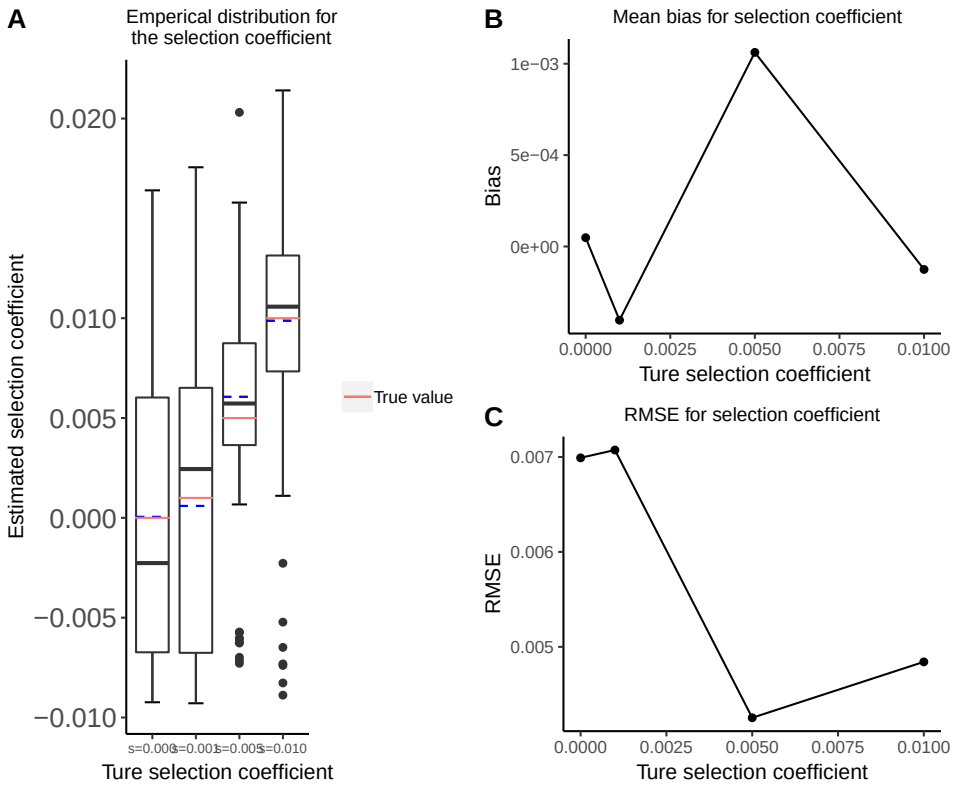

Figure 3.7: Simulation study of selection coefficient $s_{\mathscr{A}}$, the blue dash line in boxplot is the mean value for all replicates and the black solid line is the median value for all replicates.

presented them in the last column.

$(s_{\mathscr{A}} \times 10^{-3}, t_{\mathscr{A}})$	replicates	mean of MAP $\times 10^{-3}$	Bias $\times 10^{-3}$	RMSE
(0, -200)	322	0.048	0.048	0.006
(1, -200)	246	0.596	-0.496	0.007
(5, -200)	241	6.062	1.062	0.004
(10, -200)	238	9.874	-0.125	0.005

Table 3.1: Simulation study results for parameter $s_{\mathscr{A}}$

$(s_{\mathscr{A}} \times 10^{-3}, t_{\mathscr{A}})$	replicates	mean of MAP	Bias	RMSE
(0, -200)	322	-194.088	5.911	204.012
(1, -200)	246	-198.461	1.538	218.824
(5, -200)	241	-179.474	20.525	133.494
(10, -200)	238	-156.735	43.264	125.354

Table 3.2: Simulation study results for parameter $t_{\mathscr{A}}$

In histogram figure 3.6, all the true values are between the first quartile and third quartile which suggests our method is effective to jointly estimate the selection coefficient and allele age.

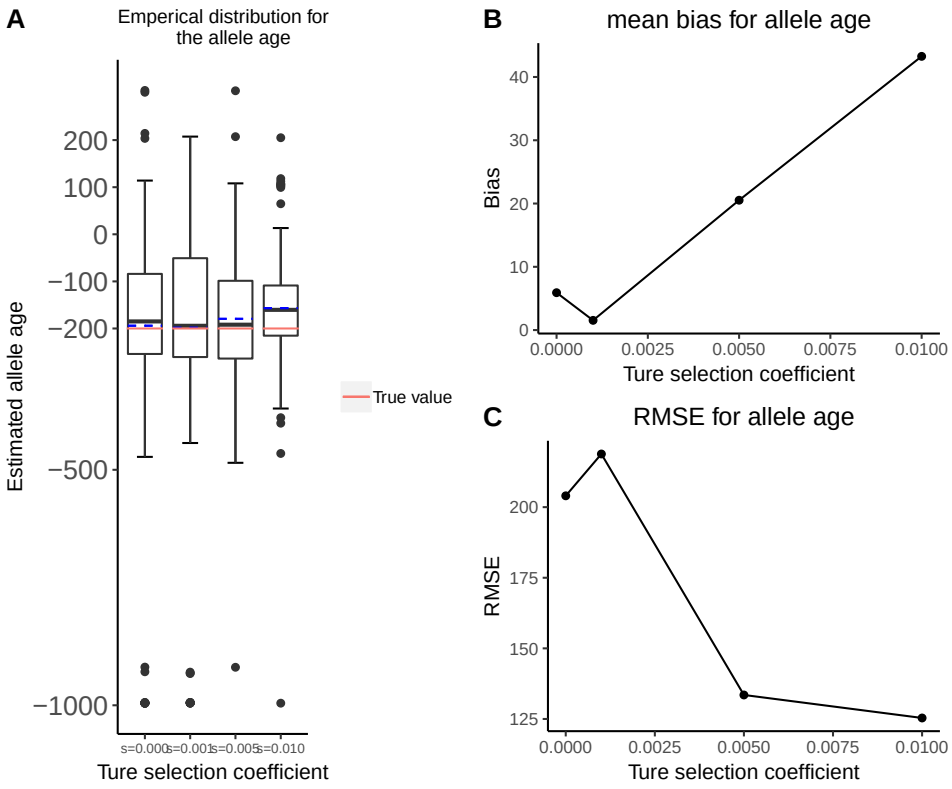

Figure 3.8: Simulation study of allele age $t_{\mathscr{A}}$, the blue dash line in boxplot is the mean value for all replicates and the black solid line is the median value for all replicates.

When selection is neutral or very small, i.e., the first two parameter sets of out trials, although the MAP estimates of selection coefficient are not centralized at zero, the mean of the MAP estimates are very close to the true value. The multi-modal distribution results from the condition of the observed mutant allele frequency are non-extinction at the first sampling time point. I added this condition to make sure the observed dataset is not all zero, otherwise, I can not do any calculation further. However, this condition leads to the underlying allele frequency that must exist until the first sampling time point. In neutral and small selection coefficient model, after first sampling time point, due to no plenty of selection power to make all alleles survive, some trajectories with decline patterns result in a negative selection coefficient estimate, and the others, who show ascent patterns, result in a positive selection coefficient estimate. But due to the non-extinction condition, we can observe the mutant allele frequency at a first sampling time point, the underlying population allele frequency is regarded either increasing from the mutant occurred or decreasing with negative selection. It will not affect the performance of this method when dealing with real data as this unrealized problem is my utility of stochastic property to simply simulation study work and will lead to no effect on parameter inference calculation.

When the selection coefficient changes to large, the MAP estimates from our method perform

much better. I also summary the average bias and RMSE in table 3.1 and table 3.2, as we can see it shows a little positive bias when selection coefficient is 0.005. Such slight positive bias for the estimates of the selection coefficient is caused by the effect of conditioning, *i.e.*, when we generate the mutant allele frequency trajectories of the underlying population according to the Wright-Fisher model, we only pick the mutant allele frequency trajectories that survive until at least the last sampling time point. These mutant allele frequency trajectories of the underlying population increase at a slightly higher rate than the unconditioned Wright-Fisher model, hence the estimates based on these mutant allele frequency trajectories will yield a slight positive bias for the selection coefficient. Although the mean of the MAP estimates are very close to the true value when selection is neutral or relative small, there exists bias for each replicate in our trial due to the conditioning Wright-Fisher model simulation. This simulation issue affects the performance of our method especially when the selection is neutral or relative small which refers to the first two parameter sets (0, -200) and (0.001, -200). When we create the observation, we only store the samples in which the mutant count is non-zero at the last observed time point. Our method to infer the selection coefficient is without such artificial condition, which means the acceptance ration is calculated based on the observed simulated trajectory only sampled once, no matter whether it is zero or not at the last observed time point. Such simulated observation under the "survive" condition actually is unfair, so our results based on those simulated observation have a little bias. As the selection coefficients increasing, our method perform well and lead to an accurate inference of selection coefficient. When the selection coefficient increase to 0.01, such strong selection yield a relatively small bias than other sets parameters, it can be explained by the relatively smaller noise with larger selection coefficients, yielding more information for natural selection.

The result of inferring allele age is not as good as the selection coefficient especially when selection is very large. When the selection coefficient is increasing from 0.001 to 0.01, the average bias changes from 1.53 to 43.3. Although the true values are still in the first and third quartile intervals, such big bias still needs to be considered carefully. Our study to infer the selection coefficient and allele age using time-serial data are strictly dominated by the observations we have. Due to the random sampling process from the underlying allele frequency trajectories, the estimates are very sensitive to observations that are simulated by the Wright-Fisher model. Especially when we investigate the allele age which is presented in the Figure 3.6 and Figure 3.8, the true value of parameter allele age is -200, however, some trails result in positive MAP estimates of allele age $t_{\mathscr{A}}$. Such cases represent the situation where the mutant occurred in underlying allele frequency trajectories at its true allele age -200, but due to the randomness of sampling, we collect the samples with a mutant allele in the very last few samples points. The lack of information in observation affects the stability of this method and often leads to the estimate result with higher variance, such circumstance is more obvious when the selection coefficient is large. Here I list all the replicates results for different parameter sets in table 3.2, in

those trials I checked each observation mutant allele frequency trajectory, around 30% simulated observed data are zero for at least first three sampling points and just recall that we only sample at five different time points with 40 individuals each time. The lack of information in observation data is the main factor resulting in biased estimates of $\hat{\boldsymbol{\vartheta}}$ for both $(s_{\mathscr{A}}, t_{\mathscr{A}})$.

Besides the residual analysis for our estimates of $\hat{\boldsymbol{\vartheta}}$, we also want to have knowledge of how those estimates distribute. As the limit number of the replicates in this simulation study, I employ a bootstrap method to derive the RMSE of the estimates $\hat{\boldsymbol{\vartheta}}$. I use 5000 resampling steps in my bootstrap and plot the histogram of the RMSE and average bias of the MAP estimators.

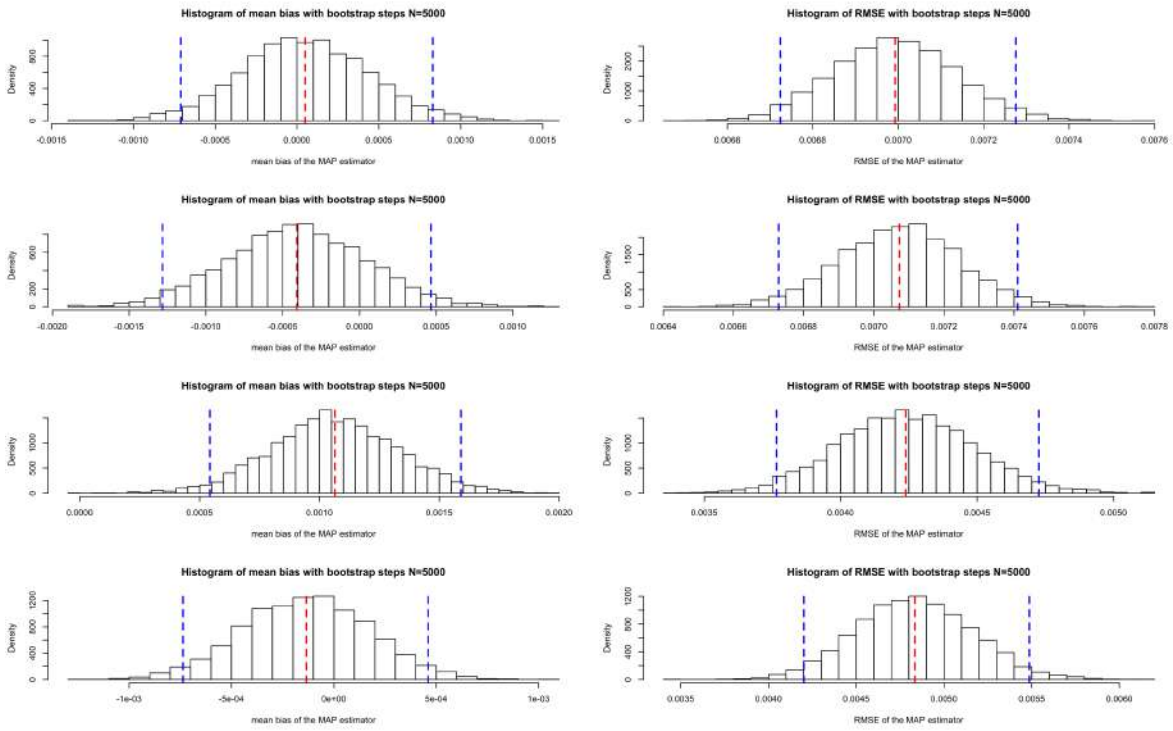

Figure 3.9: Bootstrap of RMSE and average bias for the MAP estimator of selection coefficient $s_{\mathscr{A}}$, the blue dashed lines are the 2.5 percentile and 97.5 percentile respectively . The red dashed line is the mean value for all the bootstrap resampling RMSE and average bias. From the top to bottom, each row is refer for parameter set $\left(s_{\mathscr{A}} \times 10^{-3}, t_{\mathscr{A}}\right)$ of (0, -200), (1, -200), (5, -200), (10, -200)

The bootstrap results of the average bias and the RMSE suggest similar results which I illustrate above. By using the bootstrap method, the decrease in the RMSE of the MAP estimator for the selection coefficient is more obvious when the selection coefficient increases. However, the average bias is more variable than I initially suppose. Only the parameter set (5, 200) has a very significant positive bias, and the rest of the trials do not present an obvious bias for the selection coefficient. In the bootstrap results for allele age, the RMSE results are similar to

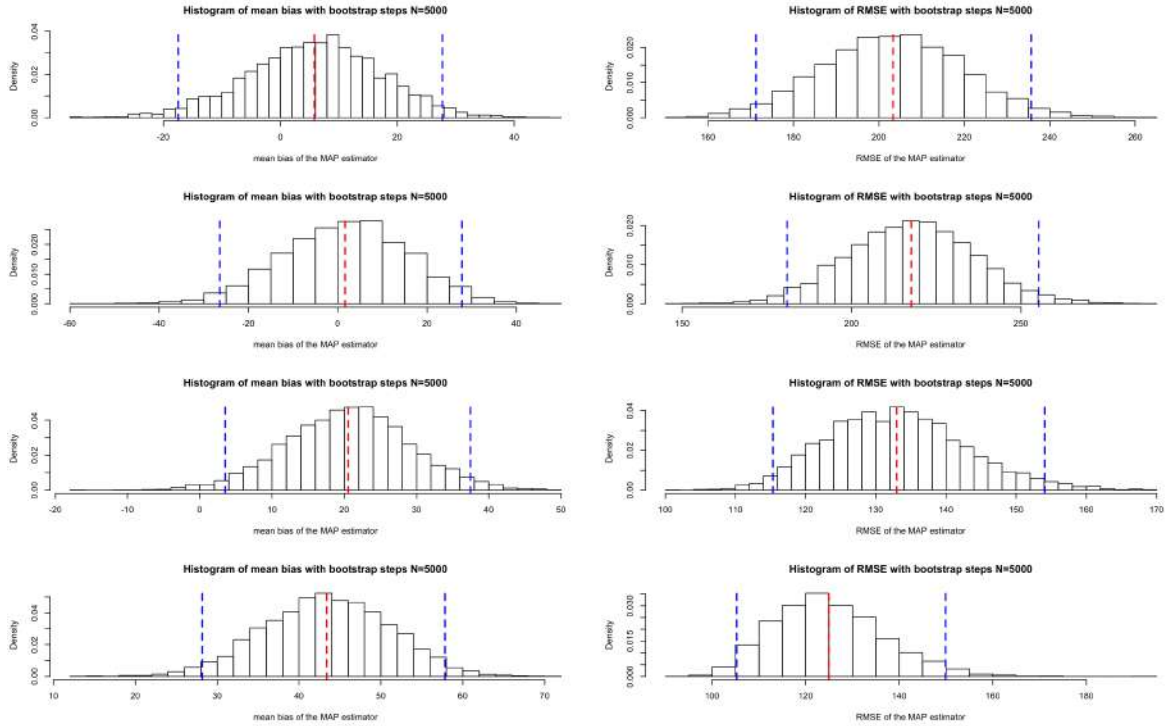

Figure 3.10: Bootstrap of RMSE and average bias for the MAP estimator of allele age $t_{\mathscr{A}}$, the blue dashed lines are the 2.5 percentile and 97.5 percentile respectively . The red dashed line is the mean value for all the bootstrap resampling RMSE and average bias. From the top to bottom, each row is refer for parameter set $\left(s_{\mathscr{A}} \times 10^{-3}, t_{\mathscr{A}}\right)$ of (0, -200), (1, -200), (5, -200), (10, -200)

$\left(s_{\mathscr{A}} \times 10^{-3}, t_{\mathscr{A}}\right)$	average bias $\times 10^{-3}$	95 % interval $\times 10^{-3}$	RMSE $\times 10^{-3}$	95 % interval $\times 10^{-3}$
(0, 200)	0.0486	[-0.7279 0.801]	6.987	[6.707, 7.278]
(1, 200)	-0.405	[-1.28 0.471]	7.070	[6.747, 7.402]
(5, 200)	1.063	[0.531, 1.583]	4.248	[3.775, 4.747]
(10, 200)	-0.126	[-0.733, 0.471]	4.828	[4.205, 5.490]

Table 3.3: Bootstrap results for parameter $s_{\mathscr{A}}$

$\left(s_{\mathscr{A}} \times 10^{-3}, t_{\mathscr{A}}\right)$	average bias	95 % interval	RMSE	95 % interval
(0, 200)	5.835	[-16.848, 26.917]	203.371	[171.956, 236.220]
(1, 200)	1.632	[-27.424, 28.394]	217.639	[179.212, 254.870]
(5, 200)	20.594	[4.024, 37.300]	133.224	[115.667, 153.881]
(10, 200)	43.123	[27.635, 57.718]	124.857	[104.455, 150.007]

Table 3.4: Bootstrap results for parameter $t_{\mathscr{A}}$

these of the selection coefficients, when the selection increases, the RMSE which reflects the standard deviation of the estimators decreases dramatically. The mean RMSE decreases from

217.7 to 124.9 when selection increases from 0.001 to 0.01, which means the variability of the MAP estimators reduces nearly by half. The bootstrap bias result is similar to the former results presented in Figure 3.8 and Table 3.2. There is a significant positive bias from our method when the selection coefficient is large, in addition to the reasons I give above, such bias may also result from the estimator we choose here. In this chapter, all the estimator I use to infer the parameters from our Bayesian framework analysis is the MAP estimator. However, as we can see from Figure 3.5, even when we have a large number of accepted samples from the PMMH, both the density curve of the selection coefficient and allele age are skewed, especially the density curve of allele age. It suggests that under such circumstances the choice of point estimate may also be one of the important reasons leading to a bias result.

3.5 Real data study

We apply our Bayesian inference procedure to real data by re-analysing the time serial sample of segregating alleles obtained via aDNA associated with horse coat colouration from Ludwig et al. [72], which has already been analysed by Ludwig et al. [72], Malaspinas et al. [74], Steinrücken et al. [110] and Schraiber et al. [99]. Ludwig et al. [72] sequenced eight loci encoding coat colour in horses for samples ranging from a pre- to a post-domestication period, which were obtained from Siberia, Middle and Eastern Europe, China and the Iberian Peninsula. In Ludwig et al. [72], the samples were grouped into six sampling time points, and by using the method of Bollback et al. [17], two of these loci, ASIP and MC1R, which showed strong fluctuations in the allele frequencies of the sample, were found to be likely to be under natural selection (see Table 4.8 for the time series data of allele frequencies for the ASIP and MC1R loci). Malaspinas et al. [74], Steinrücken et al. [110] and Schraiber et al. [99] then re-analysed the same time series data with their methods incorporating more complex demographic scenarios.

sample time (years AD)	-20000	-13100	-3700	-2800	-1100	-500
sample size	10	22	20	20	36	38
count of mutant alleles (ASIP)	0	1	15	12	15	18
count of mutant alleles (MC1R)	0	0	1	6	13	24

Table 3.5: Time series data of allele frequencies for the ASIP and MC1R loci given in Ludwig et al. [72].

I took the average length of a generation of horses to be 8 years here(see Figure 3.11 for the changes in mutant allele frequencies of the ASIP and MC1R loci), and I set the dominance parameter $h = 1$ as the mutant alleles at the ASIP and MC1R loci are both recessive [74]. Instead of trying to estimate the population size from the data itself, we run our Bayesian procedure to estimate the selection coefficient and the allele age of the mutant allele at the ASIP and MC1R loci for a few selected population sizes from 4000 to 16000, which overlaps most with the potential range for the population size in Ludwig et al. [72], since as demonstrated in Malaspinas et al.

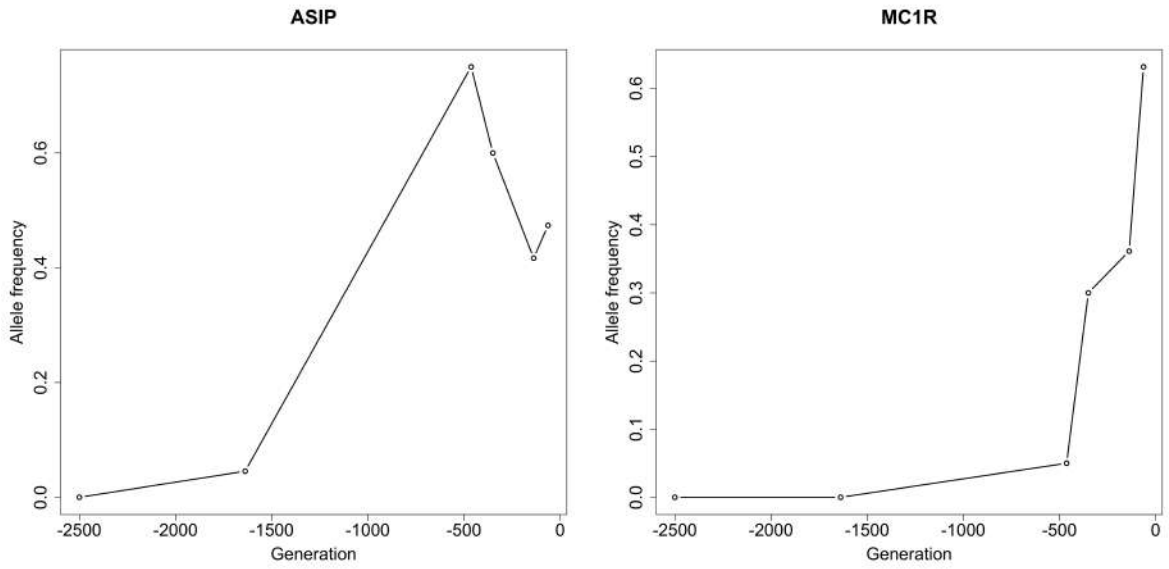

Figure 3.11: Changes in the mutant allele frequencies over time for the ASIP and MC1R loci in the sample. The average length of a generation of horses is set to be 8 years, and the sampling time points for the ASIP and MC1R loci are at generations -2500, -1637, -462, -350, -137 and -62.

[74], the sampling is not dense enough in time to give an accurate estimate of the population size. Here I tried to use various effective population size from 4000 to 32000 and generated the results in Figure 3.12 to Figure 3.13. Table 3.6 and Table 3.7 give the summary of MAP estimate for selection coefficient and allele age based on using different population size. Real data are often hard to work with. Due to the limited compute on my student account, I ran around 383 replicates inference for ASIP and 327 replicates inference for MC1R. I present the results in the histogram Figure 3.14 and summary in the Table 3.10.

population size	4000	8000	16000	32000
selection coefficient	0.00934	0.00931	0.00911	0.00893
allele age(years before present)	-5413	-5594	-5900	-5969

Table 3.6: Summary of MC1R output for different population size

population size	4000	8000	16000	32000
selection coefficient	0.00328	0.00271	0.00246	0.00229
allele age(years before present)	-16432	-17963	-19410	-20120

Table 3.7: Summary of ASIP output for different population size

From the results in the tables and figures we can see, for different population sizes, the selection coefficient changes not very significantly. The ASIP selection coefficient is around 0.0023

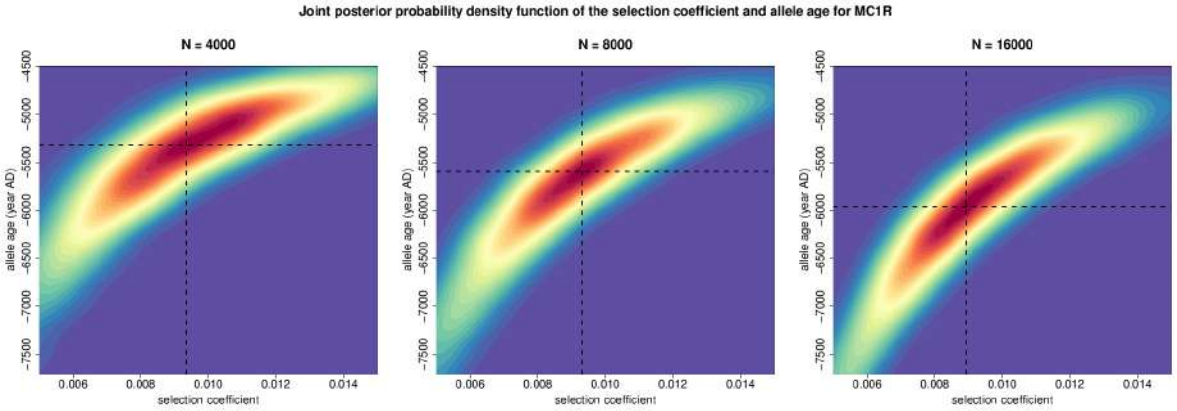

Figure 3.12: Posterior probability distributions for the selection coefficient and the allele age for the MC1R locus under different population size. The black dashed lines denote the MAP estimates of the selection coefficient and the allele age.

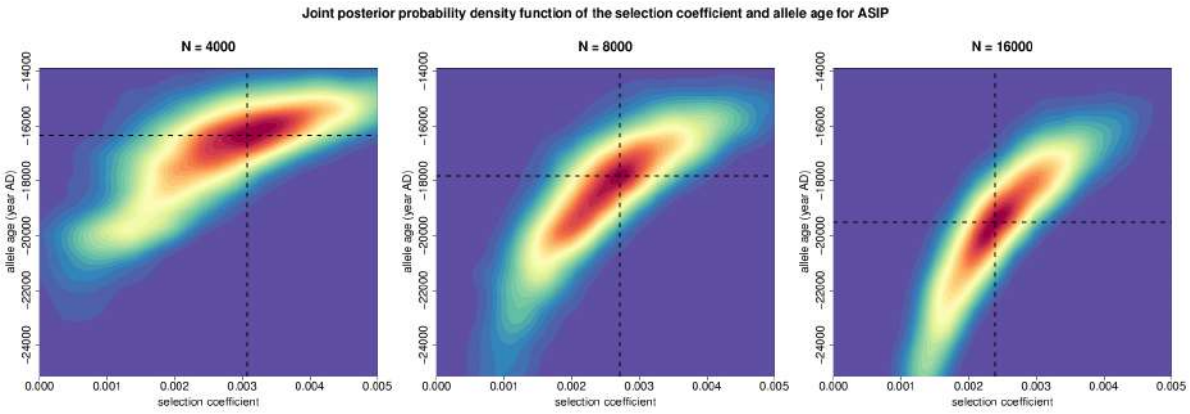

Figure 3.13: Posterior probability distributions for the selection coefficient and the allele age for the ASIP locus under different population size. The black dashed lines denote the MAP estimates of the selection coefficient and the allele age.

population size	4000	8000	16000	32000
$s_{\mathscr{A}} \times 10^{-3}$	[6.1476, 13.728]	[6.039, 12.911]	[5.849, 12.814]	[5.789, 12.451]
$t_{\mathscr{A}} \times ybp$	[-6254, -4937]	[-6431, -5119]	[-7154, -5437]	[-7311, -5571]

Table 3.8: Confidence intervals of MC1R output for different population size

to 0.0033. Only the HPD curve for population size $N = 4000$ in Figure 3.13 contain 0 which suggests the ASIP yields a slightly positive selection. In our 383 replicates inference, the 90% percentile interval is [1.619, 3.052], which also suggests the ASIP show evidence of positive selection. This result is similar to Ludwig et al. [72] who also suggest the data provided evidence

population size	4000	8000	16000	32000
$s_{\mathscr{A}} \times 10^{-3}$	[2.609, 4.051]	[1.939, 3.311]	[1.749, 3.111]	[1.689, 3.051]
$t_{\mathscr{A}} \times ybp$	[-18235, -13921]	[-19935, -15921]	[-21235, -18121]	[-23235, -18921]

Table 3.9: Confidence intervals of ASIP output for different population size

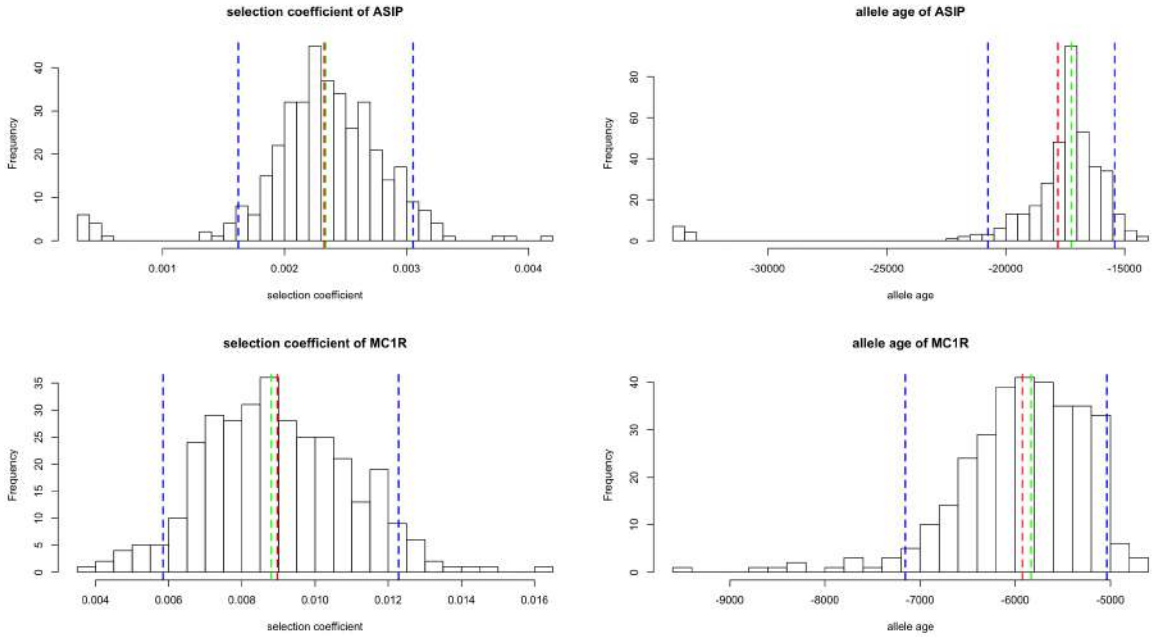

Figure 3.14: Repeated inferences results for ASIP and MC1R with fixed population sizes $N = 8000$. The blue dashed line is the 5 % percentile and 95 % percentile value, the green dashed line is the median and the red dashed line is the mean of those replicate estimates.

ASIP	mean	median	90 % percentile interval
selection $\times 10^{-3}$	2.326	2.337	[1.619, 3.052]
allele age (ybp)	-17810	-17242	[-20745, -15421]
MC1R	mean	median	90 % percentile interval
selection $\times 10^{-3}$	8.967	8.8036	[5.8476, 12.278]
allele age (ybp)	-5925	-5833	[-7154, -5037]

Table 3.10: Summary of ASIP and MC1R output for fixed $N = 8000$ with 383 and 327 replicates respectively

for slightly positive selection at the ASIP locus. For allele age, my results for different population size is from -16432 to -20120 years, and the mean of the different replicates allele age is -17810 years. The result differs from the conclusion of Malaspinas et al. [74] which suggests the age ASIP mutation is between [-20000, -13100] years with maximum likelihood estimator is -13400 years, our method suggests an earlier age of ASIP mutation. I suggest such selection on ASIP

may be explained by the changes in the environment. The mutation at locus ASIP changes the colour of the horse to black. The dark colour benefits animal to have a better shelter in the forest which can help them keep away from dangers, besides that, dark color also helps an animal with heat-absorbing which lead to a greater possibility to survive through a cold environment.

The results for analysis of the MC1R time-series data provide some evidence that the mutation at locus MC1R has a strong positive selection. For different population sizes, the estimates for the selection coefficient is around 0.01 which suggests strong selection. In 327 replicates with population size $N = 8000$, the minimum value of the MAP estimate of the selection coefficient is 0.0031 and the mean of estimators is 0.0089 with 90% percentile interval is [0.00585, 0.0123]. Ludwig et al. [72] also suggested the selection coefficient for MC1R was significant from zero and gave the explanation that such strong selection on locus MC1R may be due to selective breeding. They suggested such strong selection and the rapid change of horse color caused by the domestication which started around -5000 years. Our result of allele age also supports the Ludwig et al. [72] explanation. The estimates of allele age for different population sizes are around -5500 to -6000 which is a little earlier than domestication and the 327 replicates also have a similar result.

3.6 Dicussion

Here I present a two-step Bayesian framework for inferring both natural selection and allele age via time series data of allele frequencies. Based on adequate Wright-Fisher diffusion approximation and the PMMH Bayesian procedure, this two-step method is very stable and able to jointly estimate the selection coefficient and the allele age. The benefits of this method are obvious, under this Bayesian framework, each simulated trajectory is used to generate the posterior distribution of parameters. It enables us to avoid the discretisation of the state space in both Bollback et al. [17] and Malaspinas et al. [74] and have a continuous smooth joint density curve of selection coefficient and allele age. At the same time, unlike Steinrücken et al. [110], the two-step method based on a non-recurrent evolution model, which promises us to have a stable inference of allele age. For computational power, this method dramatically reduces the number of simulations required in the first step and the second step, which is based on the accepted simulated trajectories, also regard all the simulated trajectories valuable. Additionally, this two-step Bayesian framework is very stable to use and the intuitive idea of this method is comprehensible and easy to parallelize. However, the shortcomings of this method are also obvious. Firstly, this method is based on single locus information which means it does not take genetic recombination and the information of local linkage into account. It means we need a reliable assumption of independence before using this method.

CHAPTER 4

DETECTING AND QUANTIFYING NATURAL SELECTION AT TWO LINKED LOCI FROM TIME SERIES DATA OF ALLELE FREQUENCIES

Natural selection is a very important evolutionary process that maintains function and drives adaptation as I discussed in the former chapter. In contrast to chapter four, in this chapter, I will address a method to detecting and estimating natural selection at multiple linked loci from allele frequency time series while taking the process of genetic recombination and the information of local linkage into account. This work has arisen from a collaboration with Zhangyi He and Feng Yu. My contribution towards this work is developing a Bayesian inference method based on the hidden Markov model and particle marginal Metropolis-Hastings method, and I employ this Bayesian inference method to co-estimate the selection coefficients of two linked loci taking genetic recombination and the information of the local linkage into account. A related investigation has been submitted for Genetics.

4.1 Introduction

The recent approach of high-throughput sequencing technologies has made it possible to observe genomes in great detail over time. This provides a chance for discovering and estimating natural selection at multiple linked loci from allele frequency time series while taking the process of genetic recombination and the information of local linkage into account. Properly modelling the effects of genetic recombination and local linkage can be supposed to provide a more precise estimate for the selection coefficient and more accurate hypothesis testing on the recent action of natural selection. According to the levels of linkage disequilibrium [51], genetic recombination may either reinforce or oppose the fluctuations in allele frequencies affected by natural selection. However, with the exception of Terhorst et al. [115], all existing methods based on the Wright-

Fisher model for inferring natural selection using allele frequency time series are limited to either a single locus [*e.g.*, 17, 52, 74, 99, 110] or multiple independent loci [*e.g.*, 37, 40, 41, 103], which ignore the effect of genetic recombination effect and information from local linkage. A common approach for analysing allele frequency time series is based on the hidden Markov model (HMM) framework of Williamson and Slatkin [123], where the underlying population is assumed to evolve following the Wright-Fisher model introduced by Fisher [38] and Wright [126], and the observations are modelled by independent binomial sampling from the underlying population at each given time point [see 113, for a detailed review of the statistical inference in the Wright-Fisher model using allele frequency data]. Nevertheless, such methods are ordinarily computationally infeasible when populations become large because it requires a prohibitively large amount of computation and storage in the calculation of the likelihood. Therefore, most existing HMM-based methods are built on either the diffusion approximation of the Wright-Fisher model [*e.g.*, 17, 52, 74, 99, 110] or the moment-based approximation of the Wright-Fisher model [*e.g.*, 36, 66, 115]. Such approximations facilitate efficient integration over all possible allele frequency trajectories of the underlying population. It allows the process of likelihood computation, which is based on the observed samples, to be achieved in a moderate amount of time.

Terhorst et al. [115] extended a moment-based approximation of the Wright-Fisher model introduced by Feder et al. [36] to multiple linked loci, where the allele frequency transition between two given time points is modelled deterministically, with added Gaussian noise. To my knowledge, Terhorst et al. [115] is the only existing method for linked loci experiencing genetic drift to infer natural selection from temporal changes in allele frequencies. In Terhorst et al. [115], the underlying population dynamics at multiple linked loci was modelled using the Wright-Fisher model in their HMM framework, and the likelihood computation was carried out by approximating the Wright-Fisher model through a deterministic path with added Gaussian noise, which aims to fit a mathematically convenient transition probability density function by equating the first two moments of the Wright-Fisher model. Such a moment-based approximation works well for many applications when modelling the allele frequencies with intermediate values, but as soon as the allele frequencies get close to their boundaries (*i.e.*, allele fixation or loss), the Wright-Fisher model is poorly approximated by a Gaussian distribution that has infinite support and hence will leak probability mass into the frequency values that are larger than 1 and smaller than 0, which is not mathematically possible. This can be problematic in the inference of natural selection because natural selection is expected to rapidly drive allele frequencies towards the boundaries 0 or 1. In practice, the method of Terhorst et al. [115] is tailored toward pooled sequencing (Pool-Seq) data from evolve-and-resequence (E&R) experiments, typically on groups of up to three linked loci. Terhorst et al. [115] describe their method is designed to analyse multiple recombining sites evolving in a moderately-sized population and potentially affected by measurement error. It have shown their method is possible to detect, localize and estimate the

strength of selection in the range of $[0.01, 0.10]$ in a population of moderate size ($N \approx 10^3$) and using a moderate number (R = 3) of experimental replicates.

In this work, we propose a novel HMM-based method for Bayesian inference of natural selection at a pair of linked loci from time-series data of allele frequencies. Different from single-locus method descried in the former chapter, two-locus method accounts for the process of genetic recombination and the information of local linkage. The key innovation of our Bayesian inference procedure is that the Wright-Fisher diffusion of the stochastic evolutionary dynamics under natural selection at a pair of linked loci is used as the hidden Markov process to characterize the changes in the haplotype frequencies of the underlying population over time, which allows us to explicitly model genetic recombination and local linkage. Also, the diffusion approximation we used in our method enables us to avoid the restriction imposed by the Gaussian approximation applied in Terhorst et al. [115] that the allele frequency trajectory of the underlying population remains far away from allele fixation or loss. Genomewide scans for natural selection (GWSS), in which anomalous patterns of genetic diversity are linked to selective events, have produced a number of important results. Different from using genomewide data, the Wright-Fisher model-based methods using temporal changes in allele frequencies, which is time-series DNA data, to infer parameter of genetic interests. It can be used to make a hypothesis test whether there is a selection footprint in alleles, more importantly, it can also be used to quantitatively make an inference of the selection coefficient. The two-locus method in this chapter is the only method using time-series ancient DNA data to infer the selection coefficient at the same time accounting genetic recombination and local linkage information.

Our posterior computation is carried out with the particle marginal Metropolis-Hastings (PMMH) algorithm developed by Andrieu et al. [3], which allows for efficient calculation of the likelihood and is readily extended to model the changes in the population size and the selection coefficients as in Schraiber et al. [99]. Also, our approach can handle sampled chromosomes that contain variants with potential unknown alleles, which is common in aDNA data due to the presence of postmortem DNA damage. To illustrate the performance of our method, we run forward-in-time simulations of the two-locus Wright-Fisher model with selection. We construct two groups of simulation study to evaluate the performance of our method either with missing value or without missing value. For each group, we have 12 trials in which 6 trials are tightly linked and the other 6 trials are loosely linked. For each trial we run 100 replicates and present a box-plot for them. Besides that, to show that our method is advantageous, we make a comparison between the two-locus method and the single-locus method. The results are very obvious: our two-locus method is more promising especially in tightly linked cases. Finally, we use our method to re-analysis the ancient DNA data related to the white spotting pattern in the horse, which is known as the equine homologue of the proto-oncogene c-kit (*KIT*).

4.2 Wright-Fisher diffusion for two linked loci with selection

We begin with a short review of the Wright-Fisher diffusion for a pair of linked loci evolving under natural selection presented in He et al. [51]. This part has been mainly contributed by Zhangyi He and Feng Yu in He et al. [51].

Consider a diploid population of randomly mating individuals at a pair of linked loci $\mathscr{A}$ and $\mathscr{B}$ evolving under natural selection according to the two-locus Wright-Fisher model with selection [see, *e.g.*, 50], for which we assume discrete time and nonoverlapping generations. At each locus, there are two possible allele types, labelled $\mathscr{A}_1$, $\mathscr{A}_2$ and $\mathscr{B}_1$, $\mathscr{B}_2$, respectively, resulting in four possible haplotypes $\mathscr{A}_1\mathscr{B}_1$, $\mathscr{A}_1\mathscr{B}_2$, $\mathscr{A}_2\mathscr{B}_1$ and $\mathscr{A}_2\mathscr{B}_2$, labelled haplotypes 1, 2, 3 and 4, respectively. We attach the symbols $\mathscr{A}_1$ and $\mathscr{B}_1$ to the mutant alleles, which are assumed to arise only once in the population and be selectively advantageous, and we attach the symbols $\mathscr{A}_2$ and $\mathscr{B}_2$ to the ancestral alleles, which are assumed to originally exist in the population.

We incorporate viability selection into the population dynamics and assume that the viability is fixed from the time that the mutant allele arises and is only determined by the genotype at a single locus. More specifically, we assume that the relative viabilities of the sixteen possible (ordered) genotypes at the two loci are determined multiplicatively from the relative viabilities at individual loci based on Hardy–Weinberg equilibrium, and the relative viabilities of the three possible genotypes at each locus, *e.g.*, genotypes $\mathscr{A}_1\mathscr{A}_1$, $\mathscr{A}_1\mathscr{A}_2$ and $\mathscr{A}_2\mathscr{A}_2$ at a given locus $\mathscr{A}$, are taken to be 1, $1-h_{\mathscr{A}}s_{\mathscr{A}}$ and $1-s_{\mathscr{A}}$, respectively, where $s_{\mathscr{A}} \in [0,1]$ is the selection coefficient and $h_{\mathscr{A}} \in [0,1]$ is the dominance parameter. For example, the relative viability of the $\mathscr{A}_1\mathscr{B}_2/\mathscr{A}_2\mathscr{B}_2$ genotype is $(1-h_{\mathscr{A}}s_{\mathscr{A}})(1-s_{\mathscr{B}})$. So we can have the relative viability table as Table 4.1. We designate the recombination rate of the two loci on the same chromosome by $r \in [0, 0.5]$ and we assume that the population size is fixed to be N individuals over time.

genotypes	$\mathscr{A}_1\mathscr{B}_1$	$\mathscr{A}_1\mathscr{B}_1$	$\mathscr{A}_2\mathscr{B}_1$	$\mathscr{A}_2\mathscr{A}_2$
$\mathscr{A}_1\mathscr{B}_1$	1	$1-h_{\mathscr{B}}s_{\mathscr{B}}$	$1-h_{\mathscr{A}}s_{\mathscr{A}}$	$(-h_{\mathscr{A}}s_{\mathscr{A}})(-h_{\mathscr{B}}s_{\mathscr{B}})$
$\mathscr{A}_2\mathscr{B}_1$	$1-h_{\mathscr{B}}s_{\mathscr{B}}$	$1-s_{\mathscr{B}}$	$(-h_{\mathscr{A}}s_{\mathscr{A}})(-h_{\mathscr{B}}s_{\mathscr{B}})$	$(1-h_{\mathscr{A}}s_{\mathscr{A}})(1-s_{\mathscr{B}})$
$\mathscr{A}_1\mathscr{B}_2$	$1-h_{\mathscr{A}}s_{\mathscr{A}}$	$(-h_{\mathscr{A}}s_{\mathscr{A}})(-h_{\mathscr{B}}s_{\mathscr{B}})$	$1-s_{\mathscr{A}}$	$(1-s_{\mathscr{A}})(1-h_{\mathscr{B}}s_{\mathscr{B}})$
$\mathscr{A}_2\mathscr{B}_1$	$(-h_{\mathscr{A}}s_{\mathscr{A}})(-h_{\mathscr{B}}s_{\mathscr{B}})$	$(1-h_{\mathscr{A}}s_{\mathscr{A}})(1-s_{\mathscr{B}})$	$(1-s_{\mathscr{A}})(1-h_{\mathscr{B}}s_{\mathscr{B}})$	$(1-s_{\mathscr{A}})(1-s_{\mathscr{B}})$

Table 4.1: Summary of relative viability for 16 possible genotype combination

In this work, we consider a scaling limit of the Wright-Fisher model, where the unit of time is rescaled by $2N$. The scaled selection coefficients $\alpha_{\mathscr{A}} = 2Ns_{\mathscr{A}}$ and $\alpha_{\mathscr{B}} = 2Ns_{\mathscr{B}}$, and the scaled recombination rate $\rho = 4Nr$ are kept constant while the population size N is taken to infinity. Based on the idea of He et al. [51], as the population size goes to infinity, the haplotype frequency trajectories follow a standard diffusion limit of the two-locus Wright-Fisher model with selection [51]. The Wright-Fisher diffusion has already been successfully applied in the inference of natural selection from allele frequency time series data [*e.g.*, 17, 47, 52, 110], here we characterise it as the solution of the stochastic differential equation (SDE) instead in the work [*e.g.*, 99].

Let $X_i(t)$ be the frequency of haplotype i in the population at time t for $i = 1,2,3,4$ and designate the haplotype frequencies of the four possible types in the population by $\boldsymbol{X}(t)$, which evolves in the state space (*i.e.*, a 3-simplex)

$$\Omega_{\boldsymbol{X}} = \left\{ \boldsymbol{x} \in [0,1]^4 : \sum_{i=1}^{4} x_i = 1 \right\},$$

and satisfies the SDE in the form of He et al. [51]

$$d\boldsymbol{X}(t) = \boldsymbol{\mu}(\boldsymbol{X}(t))dt + \boldsymbol{\sigma}(\boldsymbol{X}(t))d\boldsymbol{W}(t), \qquad t \geq t_0 \tag{4.1}$$

with initial condition $\boldsymbol{X}(t_0) = \boldsymbol{x}_0$. In Equation 4.1. the drift term $\boldsymbol{\mu}(\boldsymbol{x})$ is

$$\begin{aligned}
\mu_1(\boldsymbol{x}) &= \alpha_{\mathscr{A}} x_1(x_3+x_4)[(x_1+x_2)h_{\mathscr{A}} + (x_3+x_4)(1-h_{\mathscr{A}})] \\
&\quad + \alpha_{\mathscr{B}} x_1(x_2+x_4)[(x_1+x_3)h_{\mathscr{B}} + (x_2+x_4)(1-h_{\mathscr{B}})] - \frac{\rho}{2}(x_1x_4 - x_2x_3) \\
\mu_2(\boldsymbol{x}) &= \alpha_{\mathscr{A}} x_2(x_3+x_4)[(x_1+x_2)h_{\mathscr{A}} + (x_3+x_4)(1-h_{\mathscr{A}})] \\
&\quad - \alpha_{\mathscr{B}} x_2(x_1+x_3)[(x_1+x_3)h_{\mathscr{B}} + (x_2+x_4)(1-h_{\mathscr{B}})] + \frac{\rho}{2}(x_1x_4 - x_2x_3) \\
\mu_3(\boldsymbol{x}) &= -\alpha_{\mathscr{A}} x_3(x_1+x_2)[(x_1+x_2)h_{\mathscr{A}} + (x_3+x_4)(1-h_{\mathscr{A}})] \\
&\quad + \alpha_{\mathscr{B}} x_3(x_2+x_4)[(x_1+x_3)h_{\mathscr{B}} + (x_2+x_4)(1-h_{\mathscr{B}})] + \frac{\rho}{2}(x_1x_4 - x_2x_3) \\
\mu_4(\boldsymbol{x}) &= -\alpha_{\mathscr{A}} x_4(x_1+x_2)[(x_1+x_2)h_{\mathscr{A}} + (x_3+x_4)(1-h_{\mathscr{A}})] \\
&\quad - \alpha_{\mathscr{B}} x_4(x_1+x_3)[(x_1+x_3)h_{\mathscr{B}} + (x_2+x_4)(1-h_{\mathscr{B}})] - \frac{\rho}{2}(x_1x_4 - x_2x_3),
\end{aligned} \tag{4.2}$$

the diffusion term $\boldsymbol{\sigma}(\boldsymbol{x})$ is

$$\boldsymbol{\sigma}(\boldsymbol{x}) = \begin{pmatrix}
\sqrt{x_1x_2} & \sqrt{x_1x_3} & \sqrt{x_1x_4} & 0 & 0 & 0 \\
-\sqrt{x_2x_1} & 0 & 0 & \sqrt{x_2x_3} & \sqrt{x_2x_4} & 0 \\
0 & -\sqrt{x_3x_1} & 0 & -\sqrt{x_3x_2} & 0 & \sqrt{x_3x_4} \\
0 & 0 & -\sqrt{x_4x_1} & 0 & -\sqrt{x_4x_2} & -\sqrt{x_4x_3}
\end{pmatrix}, \tag{4.3}$$

and $\boldsymbol{W}(t)$ is a six-dimensional standard Brownian motion. It should be noticed that the term $x_1x_4 - x_2x_3$ in Equation 4.1 is a measure of the linkage disequilibrium between loci $\mathscr{A}$ and $\mathscr{B}$, which quantifies the non-random association of the alleles at these two loci. See He et al. [51] for more details about the two-locus Wright-Fisher diffusion with selection.

4.3 Bayesian inference of natural selection

Let us consider a pair of linked loci $\mathscr{A}$ and $\mathscr{B}$ subject to natural selection on the same chromosome. Suppose that the observed data are sampled from the underlying population at a finite number

of distinct time points, with sampling time points labelled as $t_1 < t_2 < \ldots < t_K$. To be consistent with the Wright-Fisher diffusion time scale, the sampling time points $t_1 < t_2 < \ldots < t_K$ here are also measured in units of $2N$ generations. At the k-th sampling time point, we let $\boldsymbol{u}_k = (u_k^{\mathscr{A}}, u_k^{\mathscr{B}})$ and $\boldsymbol{v}_k = (v_k^{\mathscr{A}}, v_k^{\mathscr{B}})$ denote the counts of mutant alleles and ancestral alleles at loci $\mathscr{A}$ and $\mathscr{B}$. The total sample allele count is denoted as n_k, where $n_k \geqslant u^{\mathscr{A}} + v^{\mathscr{A}}$ and $n_k = u^{\mathscr{A}} + v^{\mathscr{A}}$ if and only if there does not exist any missing value in the observed data at locus $\mathscr{A}$. The population genetic parameters we are interested in here are the scaled selection coefficients $\alpha_{\mathscr{A}}$ and $\alpha_{\mathscr{B}}$, the dominance parameters $h_{\mathscr{A}}$ and $h_{\mathscr{B}}$, and the scaled recombination rate ρ, which are denoted by $\boldsymbol{\vartheta} = (\alpha_{\mathscr{A}}, h_{\mathscr{A}}, \alpha_{\mathscr{B}}, h_{\mathscr{B}}, \rho)$, where $\alpha_{\mathscr{A}} = 2Ns_{\mathscr{A}}$, $\alpha_{\mathscr{B}} = 2Ns_{\mathscr{B}}$ and $\rho = 4Nr$.

4.3.1 Hidden Markov model

In our HMM framework, the underlying population is assumed to evolve according to the Wright-Fisher diffusion in Equation 4.1 to Equation 4.3, and the observations are independently sampled from the underlying population at each sampling time point. In order to compute the posterior probability distribution $p(\boldsymbol{\vartheta} \mid \boldsymbol{u}_{1:K}, \boldsymbol{v}_{1:K})$, we need to integrate over all possible underlying haplotype frequency trajectories at each sampling time point Let $\boldsymbol{x}_{1:K} = (\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_K)$ denote the underlying haplotype frequency at each sampling time points $\boldsymbol{t}_{1:K}$. The posterior probability distribution for the parameter we are interested in can then be written as

$$p(\boldsymbol{\vartheta} \mid \boldsymbol{u}_{1:K}, \boldsymbol{v}_{1:K}) = \int_{\Omega_{\boldsymbol{X}}} p(\boldsymbol{\vartheta}, \boldsymbol{x}_{1:K} \mid \boldsymbol{u}_{1:K}, \boldsymbol{v}_{1:K}) d\boldsymbol{x}_{1:K}, \tag{4.4}$$

where

$$p(\boldsymbol{\vartheta}, \boldsymbol{x}_{1:K} \mid \boldsymbol{u}_{1:K}, \boldsymbol{v}_{1:K}) \propto p(\boldsymbol{\vartheta}) p(\boldsymbol{x}_{1:K} \mid \boldsymbol{\vartheta}) p(\boldsymbol{u}_{1:K}, \boldsymbol{v}_{1:K} \mid \boldsymbol{x}_{1:K}). \tag{4.5}$$

In Equation 4.5, $p(\boldsymbol{\vartheta})$ is the prior probability distribution for the population genetic quantities of interest. If prior knowledge is poor, we suggest to take a uniform prior over the parameter space. $p(\boldsymbol{x}_{1:K} \mid \boldsymbol{\vartheta})$ is the probability distribution for the underlying population haplotype frequency trajectories at the sampling time points $\boldsymbol{t}_{1:K}$, and $p(\boldsymbol{u}_{1:K}, \boldsymbol{v}_{1:K} \mid \boldsymbol{x}_{1:K})$ is the conditional probability for the observations at the sampling time points $\boldsymbol{t}_{1:K}$ given the underlying population haplotype frequency trajectories.

Since the Wright-Fisher diffusion is a Markov process, the probability distribution for the underlying population haplotype frequency trajectories at the sampling time points $\boldsymbol{t}_{1:K}$, i.e., $p(\boldsymbol{x}_{1:K} \mid \boldsymbol{\vartheta})$, can be decomposed as

$$p(\boldsymbol{x}_{1:K} \mid \boldsymbol{\vartheta}) = p(\boldsymbol{x}_1 \mid \boldsymbol{\vartheta}) \prod_{k=1}^{K-1} p(\boldsymbol{x}_{k+1} \mid \boldsymbol{x}_k; \boldsymbol{\vartheta}), \tag{4.6}$$

where $p(\boldsymbol{x}_1 \mid \boldsymbol{\vartheta})$ is the prior probability distribution for the underlying population haplotype frequencies at the initial sampling time point and can be taken to be a uniform prior over the state space $\Omega_{\boldsymbol{X}}$ if prior knowledge is poor. This is known as a flat Dirichlet distribution. The term in the

product $p(\boldsymbol{x}_{k+1} \mid \boldsymbol{x}_k; \boldsymbol{\vartheta})$ is the transition probability density of the Wright-Fisher diffusion between two consecutive sampling time points for $k = 1, 2, \ldots, K-1$. It can be obtained by numerically solving the Kolmogorov backward equation (or its adjoint) associated with the Wright-Fisher diffusion. However, this requires a fine enough discretisation of the state space $\Omega_{\boldsymbol{X}}$ and strongly depends on the underlying population genetic parameters. Additionally, numerically solving such a PDE in three dimensions for our posterior computation is computationally challenging and prohibitively expensive. We resort to an 'exact-approximate' Monte Carlo procedure [6] that involves simulating the Wright-Fisher SDE in Equation 4.1 to equation 4.3.

Given the underlying population haplotype frequency trajectories, the observations at each sampling time point are independent of each other, which means that

$$p(\boldsymbol{u}_{1:K}, \boldsymbol{v}_{1:K} \mid \boldsymbol{x}_{1:K}) = \prod_{k=1}^{K} p(\boldsymbol{u}_k, \boldsymbol{v}_k \mid \boldsymbol{x}_k), \tag{4.7}$$

where $p(\boldsymbol{u}_k, \boldsymbol{v}_k \mid \boldsymbol{x}_k)$ is the conditional probability for the observations at the k-th sampling time point given the haplotype frequency trajectories of the underlying population for $k = 1, 2, \ldots, K$. To calculate the emission probability $p(\boldsymbol{u}_k, \boldsymbol{v}_k \mid \boldsymbol{x}_k)$, we let $\boldsymbol{z}_k = (z_{1,k}, z_{2,k}, z_{3,k}, z_{4,k})$ denote the counts of the $\mathscr{A}_1\mathscr{B}_1$, $\mathscr{A}_1\mathscr{B}_2$, $\mathscr{A}_2\mathscr{B}_1$ and $\mathscr{A}_2\mathscr{B}_2$ haplotypes in the sample at the k-th sampling time point, which are usually unobserved. Then we have

$$p(\boldsymbol{u}_k, \boldsymbol{v}_k \mid \boldsymbol{x}_k) = \sum_{\boldsymbol{z}_k \in \Omega_{\boldsymbol{Z}_k}} p(\boldsymbol{z}_k \mid \boldsymbol{x}_k) p(\boldsymbol{u}_k, \boldsymbol{v}_k \mid \boldsymbol{z}_k), \tag{4.8}$$

where

$$\Omega_{\boldsymbol{Z}_k} = \left\{ \boldsymbol{z}_k \in \mathbb{N}^4 : \sum_{i=1}^{4} z_{i,k} = n_k, u_k^{\mathscr{A}} \leq z_{1,k} + z_{2,k} \leq n_k - v_k^{\mathscr{A}}, u_k^{\mathscr{B}} \leq z_{1,k} + z_{3,k} \leq n_k - v_k^{\mathscr{B}} \right\}. \tag{4.9}$$

Conditional on the haplotype frequency trajectories of the underlying population at the k-th sampling time point, the haplotype counts of the sample can be modelled through multinomial sampling from the underlying population with sample size n_k. We can then formulate the first term in the summation of Equation 4.8 as

$$p(\boldsymbol{z}_k \mid \boldsymbol{x}_k) = \frac{n_k!}{\prod_{i=1}^{4} z_{i,k}!} \prod_{i=1}^{4} x_{i,k}^{z_{i,k}}. \tag{4.10}$$

The second term in the summation of Equation 4.8 can be decomposed as

$$p(\boldsymbol{u}_k, \boldsymbol{v}_k \mid \boldsymbol{z}_k) = p(u_k^{\mathscr{A}}, v_k^{\mathscr{A}} \mid \boldsymbol{z}_k) p(u_k^{\mathscr{B}}, v_k^{\mathscr{B}} \mid \boldsymbol{z}_k). \tag{4.11}$$

Let ϕ denote the probability that a sampled chromosome at a single locus is of unknown type, which we assume to be identical for all loci. We therefore have

$$p(u_k^{\mathscr{A}}, v_k^{\mathscr{A}} \mid \boldsymbol{z}_k) = b(u_k^{\mathscr{A}}; z_{1,k} + z_{2,k}, 1-\phi) b(v_k^{\mathscr{A}}; z_{3,k} + z_{4,k}, 1-\phi) \tag{4.12}$$

$$p(u_k^{\mathscr{B}}, v_k^{\mathscr{B}} \mid \boldsymbol{z}_k) = b(u_k^{\mathscr{B}}; z_{1,k} + z_{3,k}, 1-\phi) b(v_k^{\mathscr{B}}; z_{2,k} + z_{4,k}, 1-\phi), \tag{4.13}$$

where

$$b(x;n,p) = \frac{n!}{x!(n-x)!} p^x (1-p)^{n-x} \tag{4.14}$$

is the binomial distribution. The probability that the sampled chromosome at a specific locus is of unknown type is usually unavailable, but we can estimate it with

$$\hat{\phi} = 1 - \frac{\sum_{k=1}^{K} (u_k^{\mathscr{A}} + v_k^{\mathscr{A}}) + \sum_{k=1}^{K} (u_k^{\mathscr{B}} + v_k^{\mathscr{B}})}{2\sum_{k=1}^{K} n_k}. \tag{4.15}$$

4.3.2 Particle marginal Metropolis-Hastings

To obtain the marginal posterior $p(\boldsymbol{\vartheta} \mid \boldsymbol{u}_{1:K}, \boldsymbol{v}_{1:K})$, we resort to MCMC techniques since the posterior probability distribution in Equation 4.3 is unavailable in a closed form. We devise a Metropolis-Hastings (MH) scheme to explore the population genetic quantities of interest with a fairly arbitrary proposal probability distribution, *e.g.*, a random walk proposal, where a sample of new candidates of the parameters $\boldsymbol{\vartheta}^\star$ is drawn from the proposal $q(\boldsymbol{\vartheta}^\star \mid \boldsymbol{\vartheta})$ and is accepted with the Metropolis-Hastings ratio

$$A = \frac{p(\boldsymbol{\vartheta}^\star)}{p(\boldsymbol{\vartheta})} \frac{p(\boldsymbol{u}_{1:K}, \boldsymbol{v}_{1:K} \mid \boldsymbol{\vartheta}^\star)}{p(\boldsymbol{u}_{1:K}, \boldsymbol{v}_{1:K} \mid \boldsymbol{\vartheta})} \frac{q(\boldsymbol{\vartheta} \mid \boldsymbol{\vartheta}^\star)}{q(\boldsymbol{\vartheta}^\star \mid \boldsymbol{\vartheta})}. \tag{4.16}$$

Now our core problem is reduced to calculating the intractable marginal likelihood $p(\boldsymbol{u}_{1:K}, \boldsymbol{v}_{1:K} \mid \boldsymbol{\vartheta})$ in Equation 4.16, which can be formulated as

$$p(\boldsymbol{u}_{1:K}, \boldsymbol{v}_{1:K} \mid \boldsymbol{\vartheta}) = \int_{\Omega_{\boldsymbol{X}}} p(\boldsymbol{x}_{1:K} \mid \boldsymbol{\vartheta}) p(\boldsymbol{u}_{1:K}, \boldsymbol{v}_{1:K} \mid \boldsymbol{x}_{1:K}) d\boldsymbol{x}_{1:K} \tag{4.17}$$

and achieved with a Monte Carlo (MC) estimate [4, 11]. This pseudo-marginal MCMC algorithm, which I have illustrated in Chapter 3, exploits the fact that the MC estimate of the marginal likelihood $p(\boldsymbol{u}_{1:K}, \boldsymbol{v}_{1:K} \mid \boldsymbol{\vartheta})$ is unbiased (or has a constant bias independent of the parameters $\boldsymbol{\vartheta}$) and targets the marginal posterior $p(\boldsymbol{\vartheta} \mid \boldsymbol{u}_{1:K}, \boldsymbol{v}_{1:K})$.

Here, we adopt a closely related approach developed by Andrieu et al. [3], which obtains an unbiased sequential Monte Carlo (SMC) estimate of the marginal likelihood $p(\boldsymbol{u}_{1:K}, \boldsymbol{v}_{1:K} \mid \boldsymbol{\vartheta})$ and targets the joint posterior $p(\boldsymbol{\vartheta}, \boldsymbol{x}_{1:K} \mid \boldsymbol{u}_{1:K}, \boldsymbol{v}_{1:K})$. This method is called particle marginal Metropolis-Hastings (PMMH) and deliveries a joint update of the population genetic quantities of interest $\boldsymbol{\vartheta}$ and the latent population haplotype frequency trajectories $\boldsymbol{x}_{1:K}$. The co-estimation of the haplotype frequency trajectories of the underlying population is interesting in its own right, but our interest here lies only in the population genetic parameters. Therefore we employ a special case of the PMMH algorithm where we do not nee to generate and save the underlying population haplotype frequency trajectories in the state of the Markov chain. Full details about the PMMH algorithm can be found in Andrieu et al. [3]. Fearnhead and Künsch [34] provided a detailed review of MC methods for estimating parameters in the HMM based on the particle filter.

In our Bayesian inference procedure, the implementation of the PMMH algorithm requires the SMC estimate of the marginal likelihood in Equation 4.16. This can be achieved with the bootstrap particle filter introduced by Gordon et al. [45]. More specifically, we first draw a sample of initial candidates of the parameters $\boldsymbol{\vartheta}$ from the prior $p(\boldsymbol{\vartheta})$, then we run a bootstrap particle filter with the proposed parameters $\boldsymbol{\vartheta}$ to obtain the SMC estimate of the marginal likelihood, denoted by $\hat{p}(\boldsymbol{u}_{1:K}, \boldsymbol{v}_{1:K} \mid \boldsymbol{\vartheta})$, which is equal to the product of average weights of all particles at the sampling time points $\boldsymbol{t}_{1:K}$. In the bootstrap particle filter, we generate particles from the Wright-Fisher diffusion using the Euler-Maruyama method which is similar to the process introduced in Chapter 4. We repeat the following steps until a sufficient number of samples of the parameters $\boldsymbol{\vartheta}$ have been obtained:

Step 1: Draw a sample of new candidates of the parameters $\boldsymbol{\vartheta}^\star$ from the proposal $q(\boldsymbol{\vartheta}^\star \mid \boldsymbol{\vartheta})$.

Step 2: Run a bootstrap particle filter with the proposed parameters $\boldsymbol{\vartheta}^\star$ to obtain the SMC estimate of the marginal likelihood $\hat{p}(\boldsymbol{u}_{1:K}, \boldsymbol{v}_{1:K} \mid \boldsymbol{\vartheta}^\star)$.

Step 3: Accept the proposed parameters $\boldsymbol{\vartheta}^\star$ with the Metropolis-Hastings ratio

$$A = \frac{p(\boldsymbol{\vartheta}^\star)}{p(\boldsymbol{\vartheta})} \frac{\hat{p}(\boldsymbol{u}_{1:K}, \boldsymbol{v}_{1:K} \mid \boldsymbol{\vartheta}^\star)}{\hat{p}(\boldsymbol{u}_{1:K}, \boldsymbol{v}_{1:K} \mid \boldsymbol{\vartheta})} \frac{q(\boldsymbol{\vartheta} \mid \boldsymbol{\vartheta}^\star)}{q(\boldsymbol{\vartheta}^\star \mid \boldsymbol{\vartheta})}. \tag{4.18}$$

Once enough samples of the parameters $\boldsymbol{\vartheta}$ have been obtained, we can compute the posterior $p(\boldsymbol{\vartheta} \mid \boldsymbol{u}_{1:K}, \boldsymbol{v}_{1:K})$ from the samples of the parameters $\boldsymbol{\vartheta}$ using nonparametric density estimation techniques [see 56, for a detailed review] and achieve the maximum a posteriori probability (MAP) estimates for the population genetic quantities of interest, defined by

$$\hat{\boldsymbol{\vartheta}}_{MAP} = \arg\max_{\boldsymbol{\vartheta}} p(\boldsymbol{\vartheta} \mid \boldsymbol{u}_{1:K}, \boldsymbol{v}_{1:K}). \tag{4.19}$$

In Chapter 4, we have a question whether the choice of estimator will affect our performance significantly when we analysis the results from the single-locus simulation study. Here we can get the minimum mean square error (MMSE) estimates alternatively for the population genetic quantities of interest to make a comparison and the MMSE is defined by

$$\hat{\boldsymbol{\vartheta}}_{MMSE} = \mathbb{E}(\boldsymbol{\vartheta} \mid \boldsymbol{u}_{1:K}, \boldsymbol{v}_{1:K}) = \int \boldsymbol{\vartheta} p(\boldsymbol{\vartheta} \mid \boldsymbol{u}_{1:K}, \boldsymbol{v}_{1:K}) d\boldsymbol{\vartheta}. \tag{4.20}$$

4.4 Simulation study of two-locus method

We run forward-in-time simulations of the two-locus Wright-Fisher model with selection and evaluate the performance of our approach on these replicate simulations by examining the bias and the root mean square error (RMSE) of our Bayesian estimates. For each simulated

dataset, given the values of the population genetic parameters $\boldsymbol{\vartheta}$ and the initial population haplotype frequencies $\boldsymbol{x}_0$, we simulate the haplotype frequency trajectories of the underlying population according to the two-locus Wright-Fisher model with selection. We take the dominance parameters to be $h_{\mathscr{A}} = 0.5$ and $h_{\mathscr{B}} = 0.5$ (*i.e.*, the heterozygous fitness is the arithmetic average of the homozygous fitness, called genic selection) and choose a population size of $N = 5000$ unless otherwise noted. After obtaining the simulated population haplotype frequency trajectories, we draw the unobserved sample haplotype counts independently at each sampling time point according to the multinomial distribution in Equation 4.10 first and then we generate the observed sample mutant allele counts and ancestral allele counts with Equation 4.11 to Equation 4.14. For simplicity we use fixed dominance parameters and population size, i.e., $h_{\mathscr{A}} = 0.5$, $h_{\mathscr{B}} = 0.5$ and $N = 5000$, and vary selection coefficients with $s_{\mathscr{A}} \in \{0.003, 0.01\}$ and $s_{\mathscr{B}} \in \{0, 0.002, 0.008\}$, and recombination rate with $r \in \{0.00001, 0.01\}$. As the dominate parameter can be regarded as one of the parameters of genetic interests to be jointly inferred in this method, we can pick any value of the dominate parameter to do this simulation study without affecting the stability of our method. Here to be simplified, we pick a fixed value of dominate parameter and effective population size to present how our method performance with different selection coefficient parameter under tightly or loosely linked cases using missing and un-missing dataset. The conclusions hold for any other values of the dominance parameters $h_{\mathscr{A}}, h_{\mathscr{B}} \in [0,1]$ and the population size $N \in \mathbb{N}$.

We run 100 replicates for each of the 12 possible parameter sets of selection coefficients and recombination rate. For each replicate, we take the same initial population haplotype frequencies to be $\boldsymbol{x}_0 = (0.04, 0.08, 0.08, 0.8)$ and simulate the haplotype frequency trajectories of the underlying population according to the two-locus Wright-Fisher model with selection. We sample 50 chromosomes from the underlying population at every 50 generations throughout 500 generations, i.e., the sampling time points are $t_1 < t_2 < \ldots < t_{10}$ and at each sampling time point the total sample allele count $n_k = 50$. Here I use the empirical frequency of MMSE estimate of selection coefficient and the calculation the proportion of the 95% highest posterior density (HPD) intervals that include the true values to present the method is stable in different parameter settings using the different category of data. As each box in one figure represent 100 replicates of the simulation study, it is hard for me to show totally 48 different parameter settings in a more detail way, so I followed the presenting method used in Malaspinas et al. [74] and Terhorst et al. [115] to use boxplot to show whether the coverage contains the true value to illustrate on the effectiveness of parameter inference.

4.4.1 Simulation study results for allele frequency data with and without missing values

The resulting box-plots of the empirical studies are shown in Figure 4.1 for the allele frequency datasets generated without missing values and Figure 4.2 for the allele frequency datasets generated with missing values ($\phi = 0.02$), respectively. The main purpose here is to show that our

method is effective and accurate for inferring the population genetic parameters based on the observation containing the missing value or not. As can be seen from the box-plot results, the MMSE estimates for the selection coefficients at both loci show little bias across the different parameter ranges, no matter whether sampled chromosomes contain unknown alleles or not, although one can discern a slight bias for small selection coefficients. With the increase of the selection coefficients, the MMSE estimates for the selection coefficients become more accurate. The bias and the RMSE of the resulting MMSE estimates listed in Table 4.2 and Table 4.3 for cases without missing value and with missing value respectively. The MAP estimate results are presented in Section 4.4.4 to be employed to illustrate how estimates choice affects the performance of our Bayesian method.

recombination rate $r = 0.00001$	Bias		RMSE	
$(s_{\mathscr{A}} \times 10^{-3}, s_{\mathscr{B}} \times 10^{-3})$	$s_{\mathscr{A}} \times 10^{-3}$	$s_{\mathscr{B}} \times 10^{-3}$	$s_{\mathscr{A}} \times 10^{-3}$	$s_{\mathscr{B}} \times 10^{-3}$
$(3,0)$	0.567	1.810	3.798	4.746
$(3,2)$	0.531	1.593	4.088	4.291
$(3,8)$	1.047	0.029	4.451	3.439
$(10,0)$	-0.890	1.861	3.084	5.016
$(10,2)$	-0.111	0.831	3.250	4.199
$(10,8)$	-0.781	0.650	3.749	3.745
recombination rate $r = 0.01$	Bias		RMSE	
$(s_{\mathscr{A}} \times 10^{-3}, s_{\mathscr{B}} \times 10^{-3})$	$s_{\mathscr{A}} \times 10^{-3}$	$s_{\mathscr{B}} \times 10^{-3}$	$s_{\mathscr{A}} \times 10^{-3}$	$s_{\mathscr{B}} \times 10^{-3}$
$(3,0)$	0.649	2.189	3.501	4.588
$(3,2)$	1.334	1.411	4.023	3.331
$(3,8)$	0.579	0.354	2.940	2.877
$(10,0)$	0.469	2.056	2.387	4.299
$(10,2)$	0.437	1.309	2.499	4.023
$(10,8)$	0.294	0.890	2.467	2.971

Table 4.2: Bias and RMSE of the MMSE estimates for 100 allele frequency datasets simulated *without* missing values across the different parameter ranges.

For each parameter set of the selection coefficients and the recombination rate, we calculate the proportion of the 95% highest posterior density (HPD) intervals that include the true values, shown in the bottom left corner of each box-plot in Figures 4.1 and Figures 4.2. On average, for the simulated datasets *without* missing values, 92% of replicates result in the true values of the selection coefficients being within their 95% HPD intervals values, comprising 93.33% for tightly linked loci and 90.67% for loosely linked loci respectively. For the simulated datasets *with* missing value, 92.08% of replicates have the true values of the selection coefficients being within their 95% HPD intervals, comprising 93.33% for tightly linked loci and 90.83% for loosely linked loci. As we can see, no matter whether the observation dataset contains the missing value or not, our method can deliver accurate estimates of selection coefficients for both loci $\mathscr{A}$ and locus $\mathscr{B}$. Besides that, another obvious finding is when the two loci are tightly linked, i.e., the

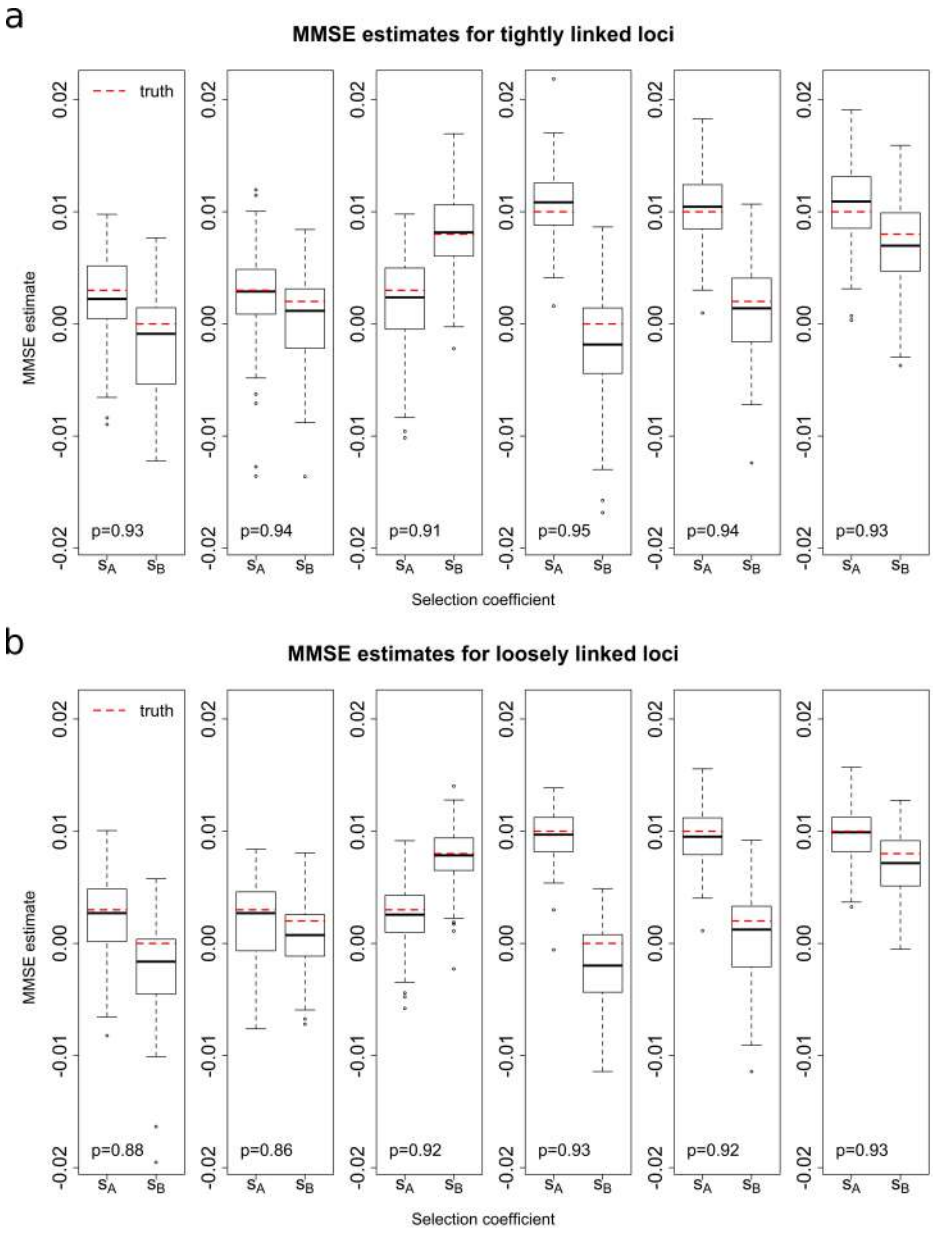

Figure 4.1: Empirical distributions of the MMSE estimates for 100 *allele frequency* datasets simulated *without* missing values. (a) Tightly linked loci with recombination rate $r = 0.00001$. (b) Loosely linked loci with the recombination rate $r = 0.01$. The p value in the bottom left corner indicates the proportion of runs where the true value of the selection coefficients falls within the 95% HPD. The tips of the whiskers denote the 2.5%-quantile and the 97.5%-quantile, and the boxes represent the first and third quartile with the median in the middle.

recombination rate is small, the performance of our method is better than they are loosely linked. Such performance gives us confidence that our method is promising and can achieve accurate inference of genetic quantities of interest.

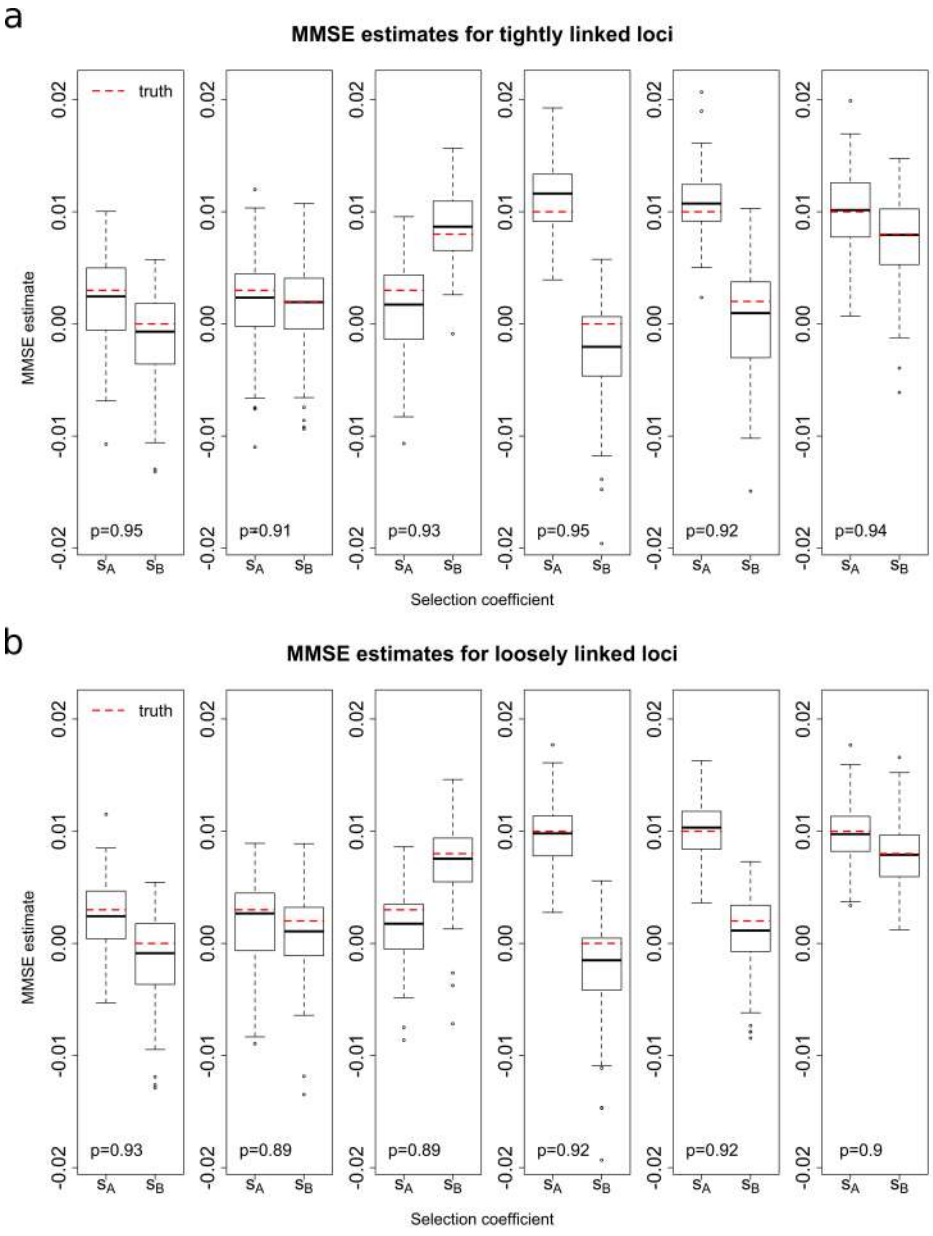

Figure 4.2: Empirical distributions of the MMSE estimates for 100 *allele frequency* datasets simulated *with* missing values. (a) Tightly linked loci with recombination rate $r = 0.00001$. (b) Loosely linked loci with the recombination rate $r = 0.01$. The p value in the bottom left corner indicates the proportion of runs where the true value of the selection coefficients falls within the 95% HPD. The tips of the whiskers denote the 2.5%-quantile and the 97.5%-quantile, and the boxes represent the first and third quartile with the median in the middle.

I also summarise the bias and RMSE for different parameter sets in Table 4.2 and Table 4.3. The first finding, easily to be observed in all trials, is the fact that when the selection coefficient $s_{\mathscr{A}}$ is fixed for 0.003 or 0.01, the larger selection coefficient of locus $\mathscr{V}$ yields smaller bias of $s_{\mathscr{B}}$.

recombination rate $r = 0.00001$	Bias		RMSE	
$(s_{\mathscr{A}} \times 10^{-3}, s_{\mathscr{B}} \times 10^{-3})$	$s_{\mathscr{A}} \times 10^{-3}$	$s_{\mathscr{B}} \times 10^{-3}$	$s_{\mathscr{A}} \times 10^{-3}$	$s_{\mathscr{B}} \times 10^{-3}$
$(3,0)$	0.943	1.389	4.097	4.500
$(3,2)$	1.246	0.582	4.949	4.088
$(3,8)$	1.758	−0.696	4.744	3.225
$(10,0)$	−1.249	2.304	3.138	5.038
$(10,2)$	−0.878	1.726	2.880	5.151
$(10,8)$	−0.107	0.481	3.700	4.108
recombination rate $r = 0.01$	Bias		RMSE	
$(s_{\mathscr{A}} \times 10^{-3}, s_{\mathscr{B}} \times 10^{-3})$	$s_{\mathscr{A}} \times 10^{-3}$	$s_{\mathscr{B}} \times 10^{-3}$	$s_{\mathscr{A}} \times 10^{-3}$	$s_{\mathscr{B}} \times 10^{-3}$
$(3,0)$	0.434	1.268	3.014	4.018
$(3,2)$	1.087	1.138	4.071	3.892
$(3,8)$	1.570	0.643	3.538	3.375
$(10,0)$	0.322	2.458	2.695	4.952
$(10,2)$	−0.044	1.344	2.622	4.485
$(10,8)$	0.271	0.108	2.663	3.015

Table 4.3: Bias and RMSE of the MMSE estimates for 100 allele frequency datasets simulated *with* missing values across the different parameter ranges.

For example, in the case of the simulated data set containing missing values with $s_{\mathscr{A}} = 0.01$, when selection coefficient $s_{\mathscr{B}}$ increases from 0 to 0.01, the bias of the MMSE estimates of $s_{\mathscr{B}}$ decreases from 2.458×10^{-3} to 0.108×10^{-3} for loosely linked and from 2.304×10^{-3} to 0.481×10^{-3} for tightly linked. Similar cases occur no matter whether the data set containing the missing value or not. It is under our expectation that the bigger value for the selection coefficient leads to more likely the simulated sampling data set reflecting the information of the underlying dynamic process. Additionally, using a more informative observation data set results in smaller RMSE of MMSE estimates for both selection coefficients on both loci. For example, in Table 4.2, the simulated data set is without missing value, when the selection coefficient is fixed as $s_{\mathscr{A}} = 0.03$, the RMSE of MMSE estimate of selection coefficient $s_{\mathscr{B}}$ reduces from 4.746×10^{-3} to 3.439×10^{-3} for tightly linked cases and he RMSE of MMSE estimate of selection coefficient $s_{\mathscr{B}}$ reduces from 4.588×10^{-3} to 2.877×10^{-3} for loosely linked. We also have the same findings when we fix the selection coefficient at locus $\mathscr{B}$ comparing with different $s_{\mathscr{A}}$ for all trials. In conclusion, for both the measurement of bias and RMSE, in comparison with weak selection, our method performs better when selection is strong.

However, in Table 4.2 and Table 4.3, we can see that when selection is weak or neutral, the estimates of selection coefficient from our method turn to a biased result which I think is very worthy of further investigation. For example, in the situation when the data is without any missing value and the selection coefficient $s_{\mathscr{A}}$ is fixed to be 0.003, the bias of natural selection on locus $\mathscr{B}$ is 1.81×10^{-3} and 2.189×10^{-3} for tightly linked and loosely linked respectively. Similarly, when the data is with missing value and the selection coefficient $s_{\mathscr{A}}$ is fixed to be

0.01, the bias of neutral selection on locus $\mathscr{B}$ is 2.304×10^{-3} and 2.458×10^{-3} for tightly linked and loosely linked respectively. That bias is significant, which suggests that our method may be unstable when the selection coefficient is very small. I suppose there are four main possible reasons which lead to this biased result; firstly, the underlying dynamic is based on the complex interplay between the four haplotype frequency $\mathscr{A}_1\mathscr{B}_1$, $\mathscr{A}_1\mathscr{B}_2$, $\mathscr{A}_2\mathscr{B}_1$ and $\mathscr{A}_2\mathscr{B}_2$, but the samples we observed in our simulation study resulting in Table 4.2, Table 4.3, Figures 4.1 and Figures 4.2 are allele frequency data. Comparing with using haplotype frequency data to make an inference of selection coefficients, the allele frequency data contains less information about the underlying population haplotype frequency trajectory and is uncertain due to the interplay between haplotypes. Additionally, it is difficult to make inferences on the selection coefficient considering recombination and local linkage, especially when selection is very small. Given that the initial mutant allele frequencies of the underlying population are taken to be close to zero in our simulation studies, the loss of the haplotypes that contain mutant alleles occurs in the early stage with high probability for weak selection. There are several simulated datasets in which the sampling frequency of the haplotype that contains mutant alleles is likely to be zero after the first few sampling time points, especially for small selection coefficients. It is more common to generate observed samples that contain some mutant allele frequency at zero for nearly all sample time points when the selection is neutral. The simulated datasets contain little information about the underlying selection coefficients. Due to the lack of information, our method is unable to make an accurate inference of the selection coefficient. Besides, in our simulation, we use the same particle number and MCMC iteration for all trials, in which the particle number is 1500 and the total number of MCMC iteration is 10^5 with the first 20% iterations are burn-in, and thinning with using 8 iterations. However, the particle number and the MCMC iteration number in our run of the PMMH may not be sufficiently large for weak selection, and this can be another potential source of the bias since weak selection can bring more uncertainties caused by genetic drift. The last reason I suspect is the choice of estimator which I used to compute MMSE in the above simulation study. Such cases also occur in Chapter 4, and here I will make a comparison between the MMSE and MAP to illustrate how the choice of point estimate affects our inference results.

4.4.2 Haplotype frequencies simulation study

To investigate how randomness resulting from the interplay between haplotypes affects the performance of our method, here I use a simulated dataset which, instead of sampling allele frequency data, consists of the observed haplotype frequency at each sample time point directly, and infer parameters with our method. The result is summarised in Table 4.4 and box-plot Figure 4.3. As I have presented in the earlier Section 4.4.1, performance of our method is effective regardless of whether the dataset contains missing values or not. The haplotype simulation study here uses the simulated data without missing values and the same random seed of the simulation

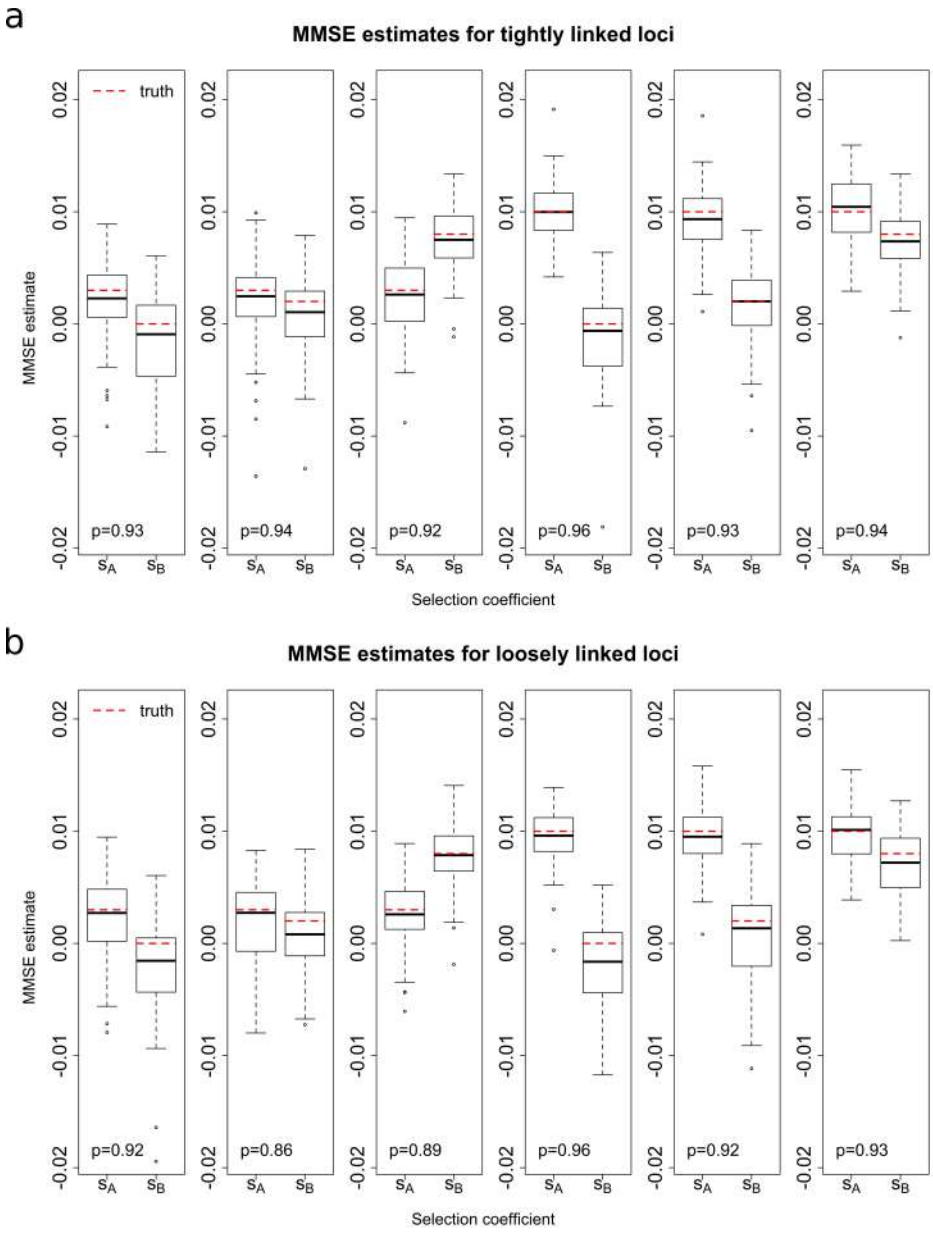

Figure 4.3: Empirical distributions of the MMSE estimates for 100 *haplotype frequency* datasets. (a) Tightly linked loci with recombination rate $r = 0.00001$. (b) Loosely linked loci with the recombination rate $r = 0.01$. The p value in the bottom left corner indicates the proportion of runs where the true value of the selection coefficients falls within the 95% HPD.

study, resulting in Table 4.2 and Figure 4.1. Compared to the estimates from allele frequency data, the estimates from haplotype frequency data are closer to their true values with smaller variances, especially for tightly linked loci. On average, 92.50% of runs result in the true values of the selection coefficients being within their 95% HPD intervals on average, with 93.67% for tightly linked loci and 91.33% for loosely linked loci. This improvement in the performance of the

recombination rate $r = 0.00001$	Bias		RMSE	
$(s_{\mathscr{A}} \times 10^{-3}, s_{\mathscr{B}} \times 10^{-3})$	$s_{\mathscr{A}} \times 10^{-3}$	$s_{\mathscr{B}} \times 10^{-3}$	$s_{\mathscr{A}} \times 10^{-3}$	$s_{\mathscr{B}} \times 10^{-3}$
$(3,0)$	0.760	1.571	3.503	4.329
$(3,2)$	0.895	1.293	3.559	3.537
$(3,8)$	0.607	0.608	3.424	3.091
$(10,0)$	−0.028	1.023	2.614	3.592
$(10,2)$	0.735	0.175	3.042	3.118
$(10,8)$	−0.239	0.647	2.876	2.680
recombination rate $r = 0.01$	Bias		RMSE	
$(s_{\mathscr{A}} \times 10^{-3}, s_{\mathscr{B}} \times 10^{-3})$	$s_{\mathscr{A}} \times 10^{-3}$	$s_{\mathscr{B}} \times 10^{-3}$	$s_{\mathscr{A}} \times 10^{-3}$	$s_{\mathscr{B}} \times 10^{-3}$
$(3,0)$	0.698	2.113	3.512	4.561
$(3,2)$	1.307	1.331	4.028	3.291
$(3,8)$	0.449	0.342	2.921	2.886
$(10,0)$	0.535	1.859	2.402	4.131
$(10,2)$	0.438	1.192	2.506	3.943
$(10,8)$	0.256	0.776	2.468	2.862

Table 4.4: Bias and RMSE of the MMSE estimates for 100 haplotype frequency datasets that we use to generate the simulated allele frequency datasets across the different parameter ranges.

estimates is to be expected as all else being equal since haplotype frequency data contain more information than allele frequency data since the complex interplay between the four haplotypes in the sample can be directly observed in haplotype frequency data but only partially observed in allele frequency data. This suggests that our method can deliver precise estimates of the selection coefficients at both loci provided that the data contain sufficient information.

4.4.3 Simulated trajectories analysis

In this section, I will discuss the bias that arises when the sampling allele frequency that contains mutant alleles is likely to be zero after the first few sampling time points, when selection coefficients are small. As I discussed briefly in Section 4.4.1, the initial mutant allele frequencies of the underlying population is close to zero when we use (0.04, 0.08, 0.08, 0.8) for haplotype frequency of $\mathscr{A}_1\mathscr{B}_1$, $\mathscr{A}_1\mathscr{B}_2$, $\mathscr{A}_2\mathscr{B}_1$ and $\mathscr{A}_2\mathscr{B}_2$ respectively. When the selection coefficient is very small, the loss of the haplotypes that contain mutant alleles is likely to occur in the early stage. To illustrate this issue, here I present all haplotype frequency trajectories, resulting in the Table 4.2, Table 4.3, Figure 4.1 and Figure 4.2.

In Figure 4.4, the plots in row (a) are for neutral selection on locus $\mathscr{B}$ and selection coefficient $s_{\mathscr{A}} = 0.003$ and the plots in row (b) are for selection coefficient $s_{\mathscr{B}} = 0.002$ and selection coefficient $s_{\mathscr{A}} = 0.003$. However, we can see from the Figure that there does not exist an obvious difference between those two groups of plots. Especially, for both sets of parameters, nearly half the haplotype trajectories for $\mathscr{A}_1\mathscr{B}_1$ show a decreasing trend to zero. In addition, haplotype trajectories $\mathscr{A}_2\mathscr{B}_1$, which contains another mutant haplotype on locus $\mathscr{B}$, also show a reduced trend. For

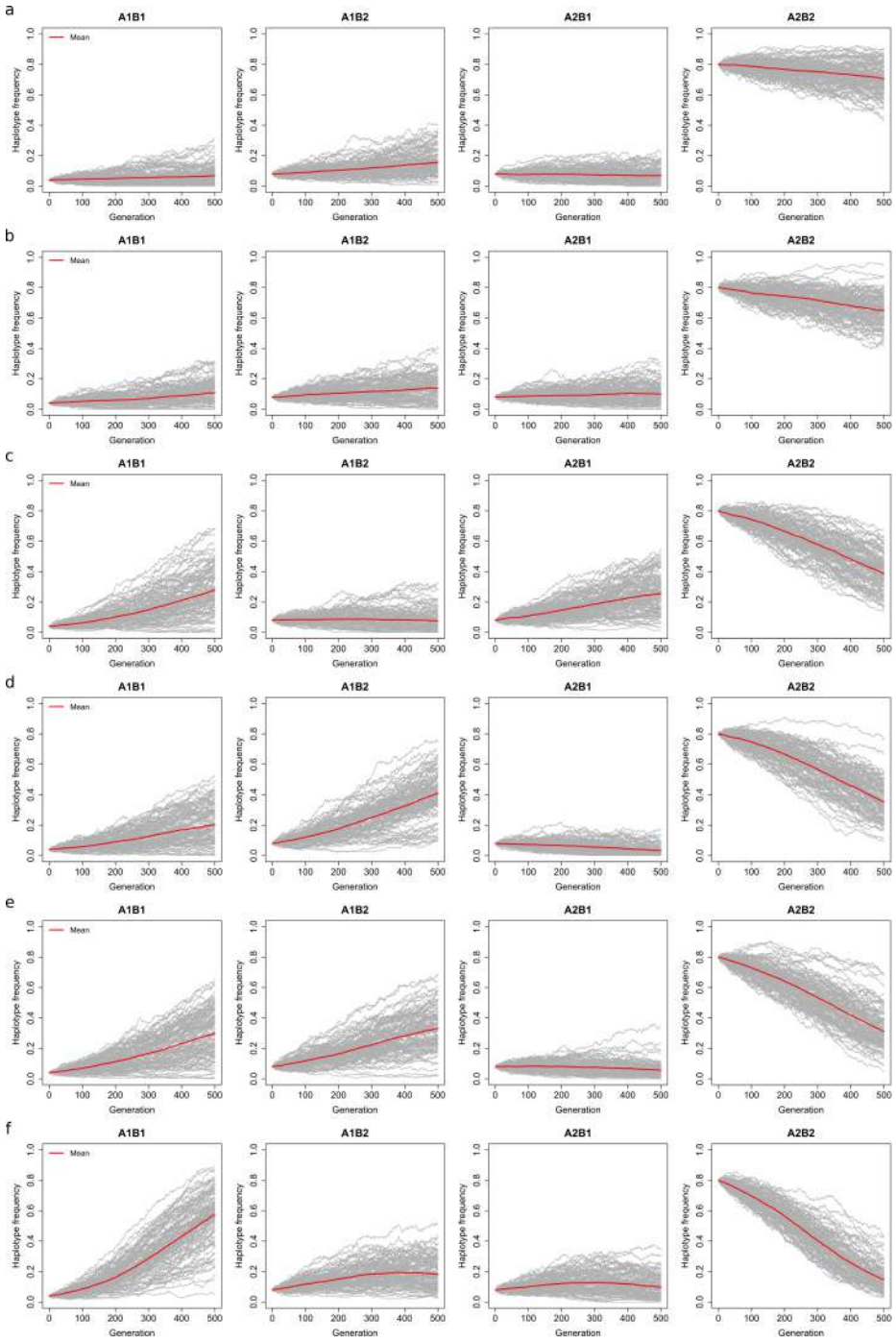

Figure 4.4: Simulated haplotype frequency trajectories of the underlying population for the allele frequency datasets simulated for the case of tightly linked loci where the recombination rate $r = 0.00001$. (a) $s_{\mathscr{A}} = 0.003$ and $s_{\mathscr{B}} = 0$. (b) $s_{\mathscr{A}} = 0.003$ and $s_{\mathscr{B}} = 0.002$. (c) $s_{\mathscr{A}} = 0.003$ and $s_{\mathscr{B}} = 0.008$. (d) $s_{\mathscr{A}} = 0.01$ and $s_{\mathscr{B}} = 0$. (e) $s_{\mathscr{A}} = 0.01$ and $s_{\mathscr{B}} = 0.002$. (f) $s_{\mathscr{A}} = 0.01$ and $s_{\mathscr{B}} = 0.008$.

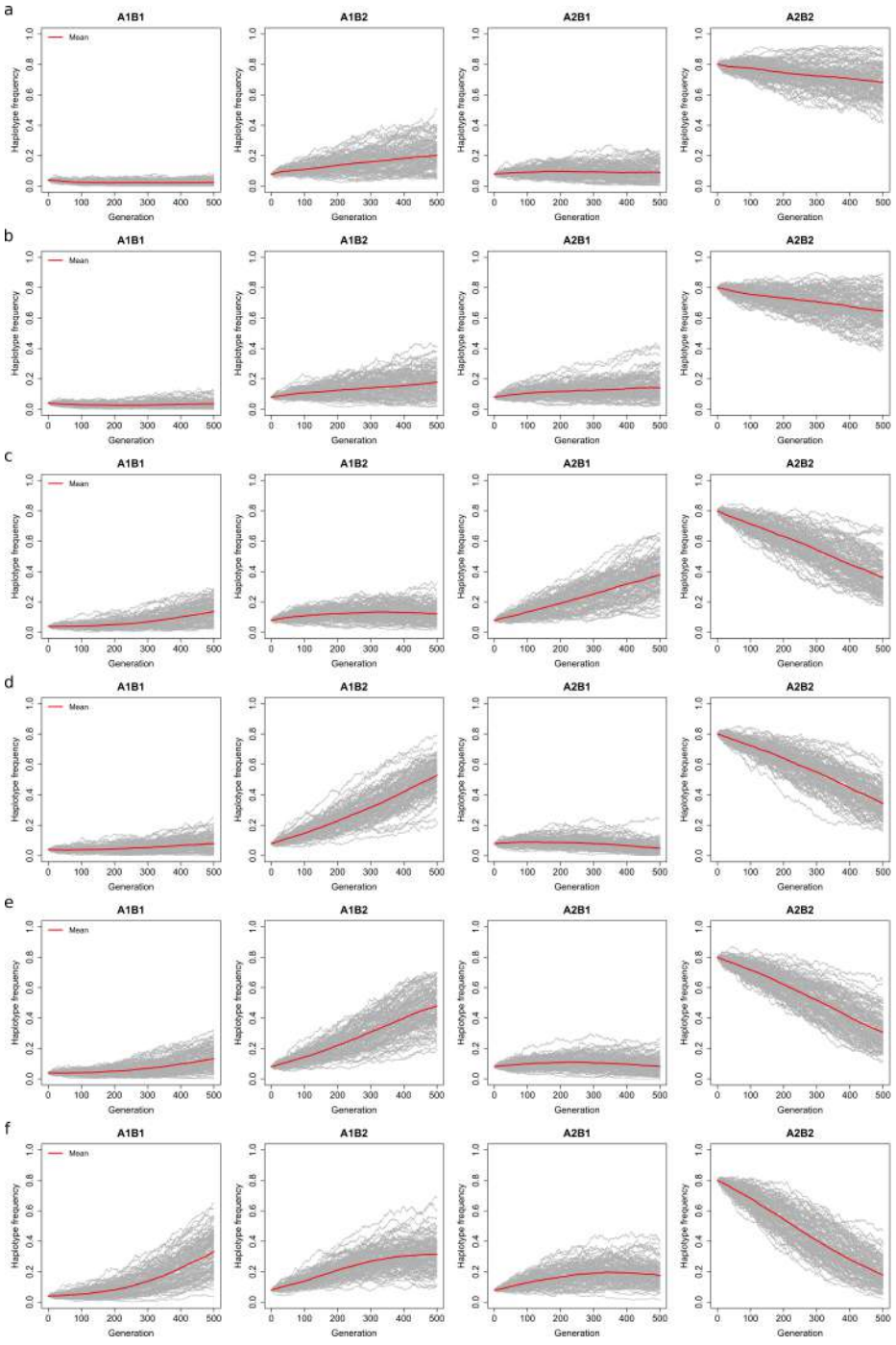

Figure 4.5: Simulated haplotype frequency trajectories of the underlying population for the allele frequency datasets simulated for the case of loosely linked loci where the recombination rate $r = 0.01$. (a) $s_{\mathscr{A}} = 0.003$ and $s_{\mathscr{B}} = 0$. (b) $s_{\mathscr{A}} = 0.003$ and $s_{\mathscr{B}} = 0.002$. (c) $s_{\mathscr{A}} = 0.003$ and $s_{\mathscr{B}} = 0.008$. (d) $s_{\mathscr{A}} = 0.01$ and $s_{\mathscr{B}} = 0$. (e) $s_{\mathscr{A}} = 0.01$ and $s_{\mathscr{B}} = 0.002$. (f) $s_{\mathscr{A}} = 0.01$ and $s_{\mathscr{B}} = 0.008$.

parameter set ($s_{\mathscr{A}} = 0.003, s_{\mathscr{B}} = 0$), haplotype trajectories $\mathscr{A}_2\mathscr{B}_1$ decreases as expected since when allele frequency $\mathscr{A}_1$ increases, the total allele frequency $\mathscr{A}_2$ decreases and in this tightly linked situation, the haplotype trajectories $\mathscr{A}_2\mathscr{B}_1$ decreases. But in comparison with the parameter set ($s_{\mathscr{A}} = 0.003, s_{\mathscr{B}} = 0.002$), we can find such a decreasing trend does not change and there are many trajectories near to zero, and some of them are relatively small after first 100 generations. The population size is $N = 5000$ and we only sample 50 chromosomes from the underlying population at each sampling time point, with such small frequencies of haplotype containing mutant alleles, we are very likely to generate observed data that does not have effective information of selection coefficient. Such cases are more obvious in Figure 4.5 where the two loci are loosely linked (recombination rate is 0.01). We can find nearly all haplotype frequency trajectories of $\mathscr{A}_1\mathscr{B}_1$ are close to zero.

By examining these haplotype trajectories, I suppose the biased inference results of selection coefficients, especially when selection is neutral and small, can be partially explained by the unfair sampling observations. To conclude, when selection is neutral and weak, we can rarely generate effective observations that reflect the mutant haplotype information of the underlying population trajectories properly in our trials, with relatively small initial mutant allele frequencies in the underlying population. Due to using the simulated data with inadequate information, sometimes our method makes a inaccurate inferences of the selection coefficient.

4.4.4 Comparing the MAP and MMSE results

In the simulation study section of Chapter 4, we suspect the point estimator may affect the performance of our method. Here I present results from the MAP estimates comparing with the MMSE estimates. For simplicity, here I just compare the results from the simulated haplotype frequency datasets which we think contain the most information. The results for using MAP as the point estimator is summarized in Table 4.5 and Figure 4.6.

recombination rate $r = 0.00001$	Bias		RMSE	
$(s_{\mathscr{A}} \times 10^{-3}, s_{\mathscr{B}} \times 10^{-3})$	$s_{\mathscr{A}} \times 10^{-3}$	$s_{\mathscr{B}} \times 10^{-3}$	$s_{\mathscr{A}} \times 10^{-3}$	$s_{\mathscr{B}} \times 10^{-3}$
$(3,0)$	0.736	1.299	3.487	4.114
$(3,2)$	0.946	1.171	3.601	3.666
$(3,8)$	0.641	0.704	3.706	3.165
$(10,0)$	0.031	1.067	2.919	3.806
$(10,2)$	0.712	0.221	3.113	3.234
$(10,8)$	-0.310	0.692	3.037	2.835
recombination rate $r = 0.01$	Bias		RMSE	
$(s_{\mathscr{A}} \times 10^{-3}, s_{\mathscr{B}} \times 10^{-3})$	$s_{\mathscr{A}} \times 10^{-3}$	$s_{\mathscr{B}} \times 10^{-3}$	$s_{\mathscr{A}} \times 10^{-3}$	$s_{\mathscr{B}} \times 10^{-3}$
$(3,0)$	0.594	1.818	3.726	4.455
$(3,2)$	1.230	1.269	4.231	3.358
$(3,8)$	0.335	0.473	3.090	3.002
$(10,0)$	0.497	1.754	2.539	4.099
$(10,2)$	0.294	1.071	2.767	3.978
$(10,8)$	0.447	0.737	2.729	2.956

Table 4.5: Bias and RMSE of the MAP estimates for 100 haplotype frequency datasets that we use to generate the simulated allele frequency datasets across the different parameter ranges.

Compared with the MMSE estimates of the selection coefficient from the haplotype frequency data, the MAP estimates have similar performance. For tightly linked data, the MAP estimates result in 93.67% of all runs having the true value within the 95% HPD intervals which is the same as the result from the MMSE estimates; similarly, for loosely linked, the MAP estimates result in 91.3% of all runs to have the true value within the 95% HPD intervals which is also the same as the result from the MMSE estimates. It means that from the boxplots comparison between the MMSE and the MAP, we can hardly detect which point estimate is more suitable.

To investigate further, I also compare the bias and the RMSE of the MAP in Table 4.5 with the MMSE results in Table 4.4. We can find the MAP estimates have similar biased results with the MMSE estimates when selection is neutral or relatively small. To be specific, when selection is strong, for example, in loosely linked simulated data with the selection coefficient $s_{\mathscr{A}} = 0.01$ and $s_{\mathscr{B}} = 0.08$, the bias for the MAP estimates of selection coefficient $s_{\mathscr{A}}$ is 0.447×10^{-3} and the bias for the MMSE estimates of selection coefficient $s_{\mathscr{A}}$ is 0.256×10^{-3} where the bias of MMSE estimates is relatively smaller than that of the MAP estimates; by contrast, when selection

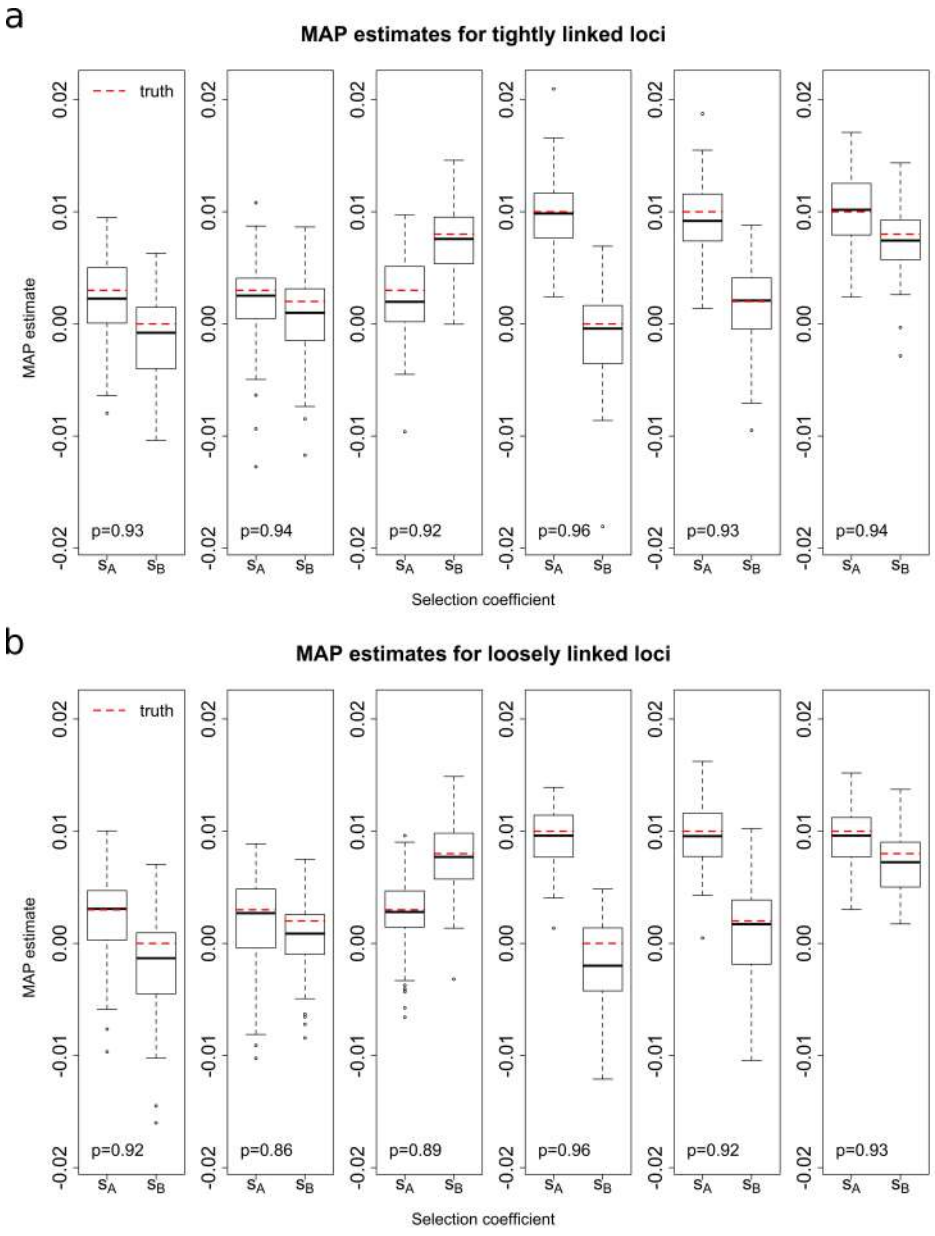

Figure 4.6: Empirical distributions of the MAP estimates for 100 haplotype frequency datasets that we use to generate the simulated allele frequency datasets across the different parameter ranges. (a) Boxplots of the MAP estimates for the case of tightly linked loci where the recombination rate $r = 0.00001$. (b) Boxplots of the MAP estimates for the case of loosely linked loci where the recombination rate $r = 0.01$.

coefficient $s_{\mathscr{A}} = 0.003$ and $s_{\mathscr{B}} = 0$, the bias for the MAP estimates of the selection coefficient $s_{\mathscr{A}}$ is 1.818×10^{-3} and the bias for the MMSE estimates of selection coefficient $s_{\mathscr{A}}$ is 2.113×10^{-3}. As we have discussed in the former section, it is more important to evaluate the performance when selection is strong, which is expected to be more informative. By comparing the bias of

the different estimates, I suggest that using the MMSE to make a point estimate will be more appropriate than using the MAP. Additionally, in comparison of the RMSE from the MMSE estimates and the RMSE from the MAP estimates, we can discover almost in all trials the MMSE estimates yield smaller RMSE than the MAP estimates which reveals the MMSE estimate is more likely to achieve a smaller variate inference of both selection coefficient $s_{\mathscr{A}}$ and selection coefficient $s_{\mathscr{B}}$. Both MMSE and MAP estimators are evaluated using the posterior probability density function(p.d.f), which here we calculated by the PMMH method. In one-dimension cases, the MMSE is the center of the mass, while the MAP is the mode of the p.d.f. In symmetric posterior p.d.f. the MMSE and MAP estimator is equal. In our method, the jointly posterior distribution of the selection coefficient is not symmetric, which results in a discrepancy between them. I can hardly decide which estimator will be the best under this circumstance, so I kept both estimator results for rest simulation studies and real data study.

4.5 Single-locus method versus two-locus method

In the last decade, there are a number of studies that have investigated selection at a single locus using time-series allele frequency data, [*e.g.*, 17, 52, 74, 99, 110]. As I illustrated in the former section, one of the biggest differences between using single-locus method and two-locus method is when we consider genetic recombination and local linkage. Here I want to show that when two loci are linked, especially tightly linked, our two-locus method is generally more accurate. All examples in this section are set with a tightly linked model with a recombination rate is $r = 0.00001$, and I use the two-locus Wright-Fisher model with selection [51] to simulate the haplotype frequency trajectories. After that, I sample 200 chromosomes from the underlying population at each sampling time point, given at generation $t_0, t_1, \cdots, t_5 := \{0, 100, 200, 300, 400, 500\}$.

4.5.1 Positively selected locus $s_{\mathscr{A}} = 0.01$ linked with a neutral locus $s_{\mathscr{B}} = 0$

In the first example, let us consider a positively selected locus $\mathscr{A}$ ($s_{\mathscr{A}} = 0.01$) linked with a neutral locus $\mathscr{B}$ ($s_{\mathscr{B}} = 0$). The initial underlying population haplotype frequencies we set as $\boldsymbol{x}_0 = (0.2, 0.1, 0.3, 0.4)$. The mutant allele frequency trajectories of the sample are shown in Figure 4.7a. As we can see, the allele frequencies of both loci display an increasing trend. I make an inference of the selection coefficient of allele $\mathscr{A}$ and allele $\mathscr{B}$ separately by using the first step of the 'two-steps' method described in Chapter 4 and present the posterior distribution of both loci in Figure 4.7b. With the single-locus method, the estimate for the selection coefficient $s_{\mathscr{A}}$ is relatively accurate, and both the MAP estimate and the MMSE estimate are close to its true value of 0.01. However, the estimate for the selection coefficient $s_{\mathscr{B}}$ is far away from its true value which is 0. We can see that the true value, which is the red dashed line in the Figure 4.7b, is even out of the 95% HPD intervals. The single-locus method shows that both the MAP and the MMSE estimate are near to around 0.005 with a 95% HPD interval only encompassing positive values.

This result is very likely to be regarded as a strong evidence for positive natural selection, even though we set the locus $\mathscr{B}$ to be neutral. In Figure 4.7c, I present the result from our two-locus method and summarise the estimates in Table 4.6. In comparison, the estimates for both of the selection coefficients $s_{\mathscr{A}}$ and $s_{\mathscr{B}}$ are fairly accurate, with the estimate of selection coefficient $s_{\mathscr{A}}$ is similar to the result from the single-locus method. For locus $\mathscr{B}$, we can see there are big differences: the estimates from the single-locus are around 0.005, whereas the estimate from the two-locus are much smaller and close its true value 0. Besides that, in comparison with the single-locus method, the 95% HPD interval result from the two-locus method is $[-0.905, 1.003]$, which is very likely to suggest the locus $\mathscr{B}$ is neutral.

		single-locus method ($\times 10^{-2}$)	two-locus method ($\times 10^{-2}$)
selection coefficient $s_{\mathscr{A}} = 0.01$	MAP	1.153	1.099
	MMSE	1.125	1.216
	95% HPD	$[0.678, 1.506]$	$[0.684, 1.837]$
selection coefficient $s_{\mathscr{B}} = 0$	MAP	0.501	0.192
	MMSE	0.507	0.125
	95% HPD	$[0.090, 0.905]$	$[-0.905, 1.003]$

Table 4.6: A comparison of the Bayesian estimates obtained by using the single-locus method and the two-locus method from the simulated dataset of a positively selected locus tightly linked with a neutral locus.

I also display the haplotype frequency trajectories in Figure 4.7d and the allele frequency trajectories in Figure 4.7e to illustrate why in such a situation the single-locus method is outperformed. The increase in the frequency of the $\mathscr{B}_1$ allele is not on selection at locus $\mathscr{B}$; it is due to the increase in haplotype frequency $\mathscr{A}_1\mathscr{B}_1$ which contains the allele $\mathscr{A}_1$ under strong selection. When $\mathscr{A}_1$ increases, $\mathscr{A}_2$ is expected to decrease; however, such a decrease in the haplotype $\mathscr{A}_2\mathscr{B}_2$ has a fast rate than the haplotype $\mathscr{A}_2\mathscr{B}_1$. Additionally, the increase in allele $\mathscr{A}_1$ mainly results from the haplotype $\mathscr{A}_1\mathscr{B}_1$ which also increases allele frequency $\mathscr{B}_1$, whereas the haplotype $\mathscr{A}_1\mathscr{B}_2$ does not change much. Finally, these interplay between all four haplotypes result in the case that the allele frequency of locus $\mathscr{B}_1$ seems to increase with time and lead the single-locus method to a positive biased result. In comparison, the two-locus method takes the interplay between all haplotypes into account and achieves accurate estimates for both selection coefficients $s_{\mathscr{A}}$ and $s_{\mathscr{B}}$.

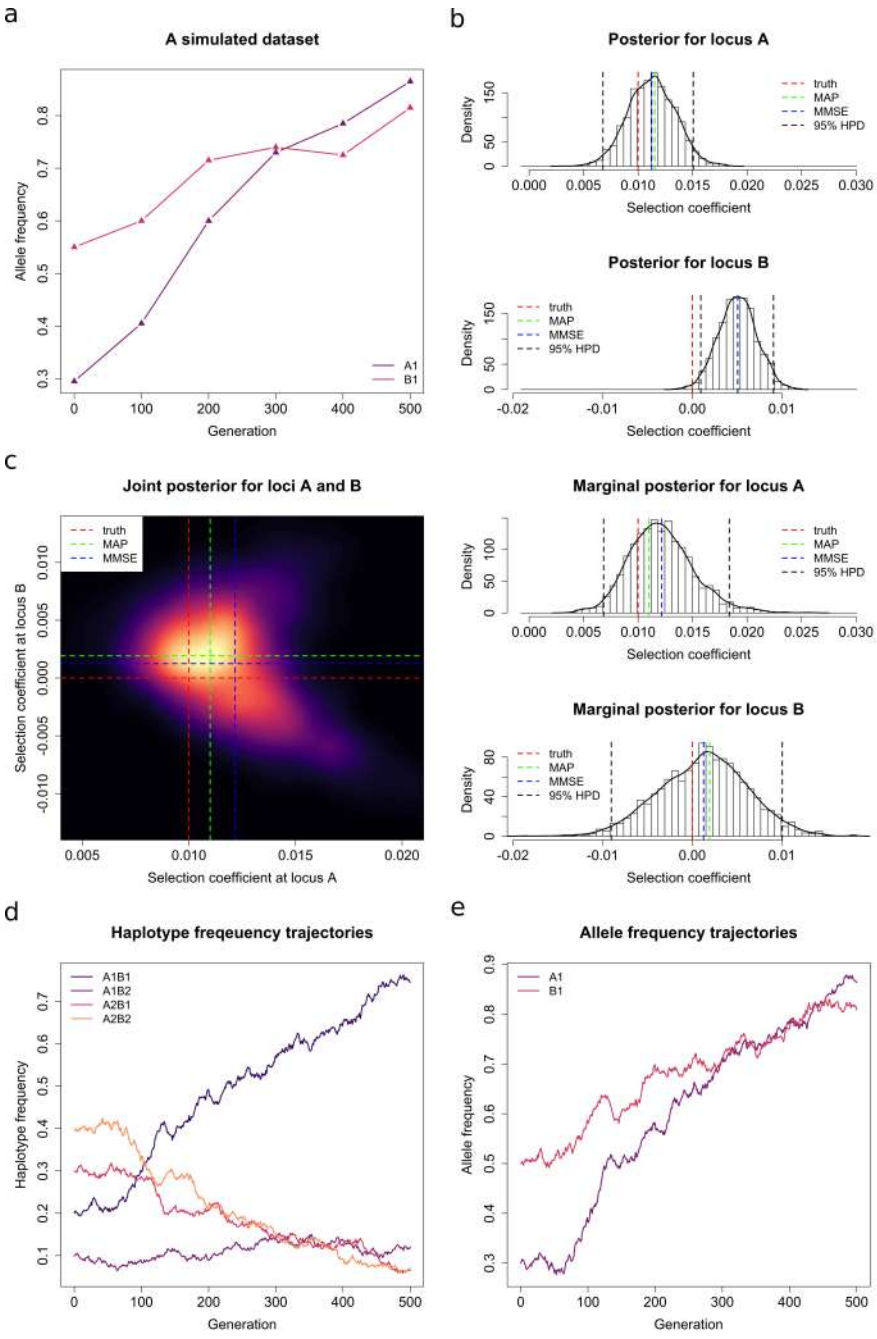

Figure 4.7: A comparison of the performance differences of the single-locus method and the two-locus method on the simulated dataset of a positively selected locus tightly linked with a neutral locus. (a) The simulated dataset. (b) Posteriors obtained with a single-locus method. (c) Posteriors obtained with a two-locus method. (d) Population mutant allele frequency trajectories. (e) Population haplotype frequency trajectories.

4.5.2 Two positively selected and tightly linked loci $s_{\mathscr{A}} = 0.01$ and $s_{\mathscr{B}} = 0.005$

In the second example, we consider two positively selected and tightly linked loci $\mathscr{A}$ and $\mathscr{B}$ with selection coefficients to be $s_{\mathscr{A}} = 0.01$ and $s_{\mathscr{B}} = 0.005$. The initial underlying population haplotype frequencies here is to be $\boldsymbol{x}_0 = (0.05, 0.05, 0.7, 0.2)$. The results are presented in Figure 4.8 and summarised in Table 4.7. In Figure 4.8a, the simulated sample allele frequencies $\mathscr{A}_1$ shows an increasing trend as expected, in contrast to allele $\mathscr{A}_1$. The sample allele frequencies $\mathscr{B}_1$ does not show a clear trend despite there being a positive selection on locus $\mathscr{B}_1$ with selection coefficient $s_{\mathscr{B}} = 0.005$. The single-locus method performs well when estimating the selection coefficient $s_{\mathscr{A}}$, but it fail to estimate the the selection coefficient $s_{\mathscr{B}}$ where the single locus method delivers an estimate for the selection coefficient $s_{\mathscr{B}}$ is roughly -0.0007 with the true value $s_{\mathscr{B}} = 0.005$, which is out of the 95% HPD intervals in Figure 4.8b. In Table 4.7, the 95% HPD intervals from the single-locus method is $[-0.510, 0.387]$. So that we can not reject the hypothesis that locus $\mathscr{B}_1$ is neutral, although in fact, the $\mathscr{B}_1$ allele is favoured by natural selection. When we turn to use the two-locus method, the MMSE estimates of selection coefficient $s_{\mathscr{A}}$ is 0.01013 and the MMSE estimates of selection coefficient $s_{\mathscr{B}}$ is 0.00423, and both of the estimates are relatively accurate comparing with single-locus method. Although the 95% HPD intervals of selection coefficient $s_{\mathscr{B}}$ still contains 0, the MMSE estimate is much close to its true value.

		single-locus method ($\times 10^{-2}$)	two-locus method ($\times 10^{-2}$)
selection coefficient $s_{\mathscr{A}} = 0.01$	MAP	0.732	0.775
	MMSE	0.739	1.013
	95% HPD	$[0.299, 1.154]$	$[0.340, 1.912]$
selection coefficient $s_{\mathscr{B}} = 0.005$	MAP	-0.117	0.194
	MMSE	-0.071	0.423
	95% HPD	$[-0.510, 0.387]$	$[-0.462, 1.531]$

Table 4.7: A comparison of the Bayesian estimates obtained by using the single-locus method and the two-locus method from the simulated dataset of a pair of positively selected and tightly linked loci.

In Figure 4.8d and Figure 4.8e, I display the allele frequencies trajectories and the haplotype frequencies trajectories to illustrate why the single-locus fails in this case.The selection coefficients for the haplotypes $\mathscr{A}_1\mathscr{B}_1$, $\mathscr{A}_1\mathscr{B}_2$, $\mathscr{A}_2\mathscr{B}_1$ and $\mathscr{A}_2\mathscr{B}_1$ are 0.015, 0.01, 0.005, 0, respectively. The initial underlying population haplotype frequencies is $\boldsymbol{x}_0 = (0.05, 0.05, 0.7, 0.2)$. The allele frequency $\mathscr{A}_1$ increases dramatically with the strongest two selection coefficient haplotypes $\mathscr{A}_1\mathscr{B}_1$ and $\mathscr{A}_1\mathscr{B}_2$. When allele $\mathscr{A}_1$ is increasing, the allele frequency $\mathscr{A}_2$ needs to decrease. Such a decrease occurs in haplotype $\mathscr{A}_2\mathscr{B}_1$ and $\mathscr{A}_2\mathscr{B}_2$ with selection coefficients 0.05 and 0. The initial haplotype frequency of $\mathscr{A}_2\mathscr{B}_1$ is 0.7, and the frequency change on haplotype $\mathscr{A}_2\mathscr{B}_1$ is more likely to affect the allele frequency of $\mathscr{B}_1$. Due to $\mathscr{B}_1$ being out-competed by $\mathscr{A}_1$, the allele frequency of $\mathscr{B}_1$ seems to be constant at all sampling time points, which leads the single-locus method to result in a biased estimate. Comparing with the single-locus method, our two-locus method can capture

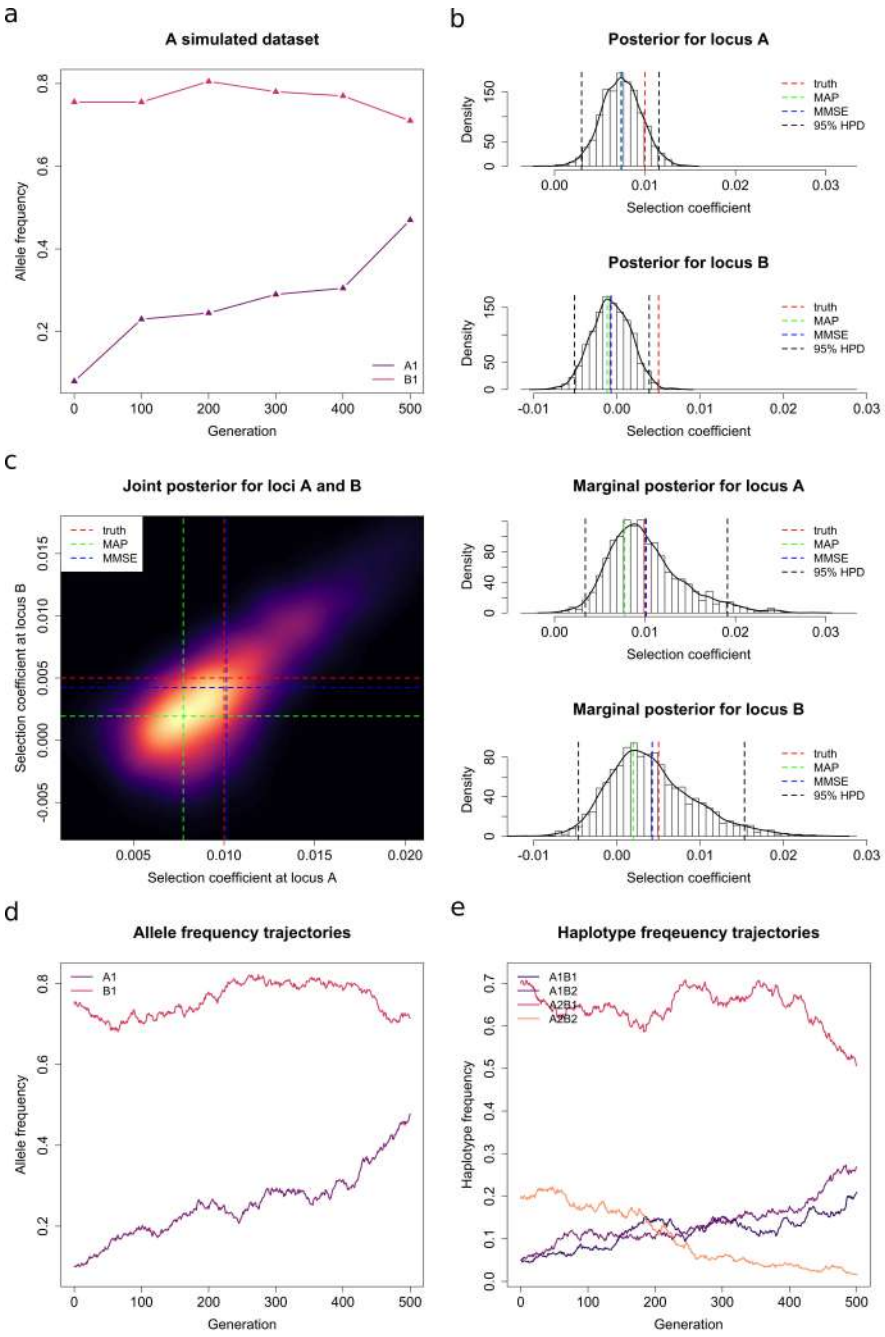

Figure 4.8: A comparison of the performance differences of the single-locus method and the two-locus method on the simulated dataset of a pair of positively selected and tightly linked loci. (a) The simulated dataset. (b) Posteriors obtained with a single-locus method. (c) Posteriors obtained with a two-locus method. (d) Population mutant allele frequency trajectories. (e) Population haplotype frequency trajectories.

more information from the complex interplay of all the underlying haplotypes, which sometimes leads a single-locus method to an unfaithful estimate. Yu and Etheridge [130] and Cuthbertson

et al. [23] also suggest when genetic recombination and the local linkage are necessary to be considered, the estimates from the single-locus method may often be inaccurate. Our two-locus method takes genetic recombination and the information of local linkage into an account yielding faithful estimates of both selection coefficients.

4.6 Analysis of real data

Let us employ this two-locus method to re-analyse the time serial samples of segregating alleles of the equine homologue of proto-oncogene c-kit (*KIT*). This data is published by previous studies of Ludwig et al. [72], Pruvost et al. [93] and Wutke et al. [128], and summarized in Table 4.8. The *KIT* gene in horses resides on the long arm of chromosome 3 and presents two intervals associated with white spotting patterns, one in the intron 13 which codes for tobiano (*KIT13*), and the other in intron 16 which codes for sabino (*KIT16*). At the *KIT13* locus, the ancestral allele is designated *KM0*, while the mutant allele, associated with the tobiano pattern and acting as dominant [19], is designated *KM1*. The tobiano pattern is characterised by depigmented patches of skin and associated hair that often cross the dorsal midline and cover the legs. At the *KIT16* locus, the ancestral allele is designated *sb1*, while the mutant allele associated with the sabino pattern and acting as semi-dominant [18], is designated *SB1*. The sabino pattern is characterised by irregularly bordered white patches of skin and associated hair that begin at the extremities and face and may extend up to the belly and midsection.

		KIT13	*KIT16*
sample time	sample size	*KM0/KM1*	*sb1/SB1*
17146	22	22/0	22/0
7029	14	14/0	14/0
5472	48	45/3	44/2
4442	24	24/0	24/0
3916	28	28/0	28/0
3352	56	53/3	52/4
2624	30	26/4	24/0
2330	14	11/3	12/0
1134	100	77/3	86/0

Table 4.8: Time serial samples of segregating alleles at the *KIT13* and *KIT16* loci. The unit of the sampling time is the year before present (BP).

We set the dominance parameters $h = 0$ for *KIT13* as the *KM1* allele is dominant, and $h = 0.5$ for *KIT16* as the *SB1* allele is semi-dominant. Following Der Sarkissian et al. [27], we take the population size to be $N = 16000$ and the average length of a generation of the horse to be 8 years, where Schraiber et al. [99] used the same population size and length of a generation in their publications. In Figure 4.9, I display all possible mutant allele frequency trajectories of the sample at the *KIT13* and *KIT16* loci in Figure 4.9 from the third sampling time point due to

the mutant allele counts are zero at the first sampling time points. As can be seen in Table 4.8, there are various sampling time points when the sequencing of the aDNA material yielded many unknown alleles at loci *KIT13* and/or *KIT16*. Wutke et al. [128] suggests that both mutant alleles, *KM1* and *SB1*, arose after the domestication of the horse, which is thought to have started in the Eurasian Steppes around 5500 years BP [89]. Therefore we discard the first two samples from our analysis.

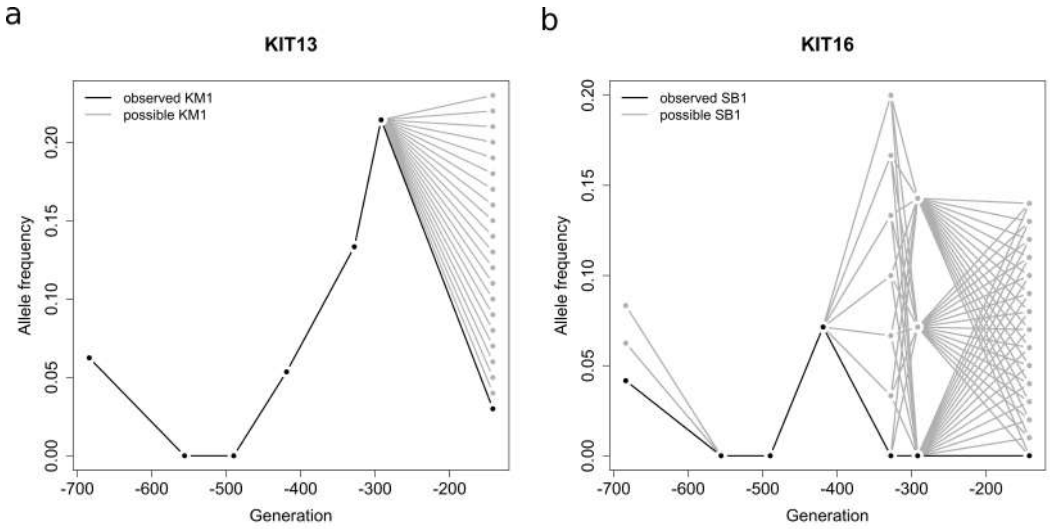

Figure 4.9: Potential changes in the mutant allele frequencies of the sample over time at the *KIT13* and *KIT16* loci. Ancient horse samples were taken at generations -684, -556, -490, -419, -328, -292 and -142. (a) Sample mutant allele frequency trajectories for *KIT13*. (b) Sample mutant allele frequency trajectories for *KIT16*.

As we can see from Figure 4.9, the *KIT* dataset contains several missing values, which leads to a difficulty to determine whether there exists an obvious increasing or decreasing trend. With those unknown values, some single-locus methods easily fails to deliver a faithful estimate. Our two-locus method has been proved is desirable to achieve a co-estimate of selection coefficients for the mutant alleles at the *KIT13* and *KIT16* loci using the sampled chromosomes containing unknown alleles. The genetic distance between the locus *KIT13* and *KIT16* is 4688 bp, here we follow the Dumont and Payseur [29] to choose a set of average rates of recombination which are 5×10^{-9}, 1×10^{-8} and 5×10^{-8} crossovers/bp to evaluate the recombination between the locus *KIT13* and *KIT16*. The results are presented in the Figure 4.10 and Table 4.9.

As can be seen, the change of the recombination rate does not affect the MMSE estimates significantly. In comparison, the MAP estimates for selection coefficient of the mutant allele on locus *KIT13* vary from 0.079×10^{-2} to -0.021×10^{-2} as recombination rate changing from 0.234×10^{-4} to 0.469×10^{-4}. This is because the MAP estimates are based on the posterior distribution approximated by nonparametric estimation techniques and the performance of using the MAP sometime is unstable due to the limited number of iteration of the PMMH process.

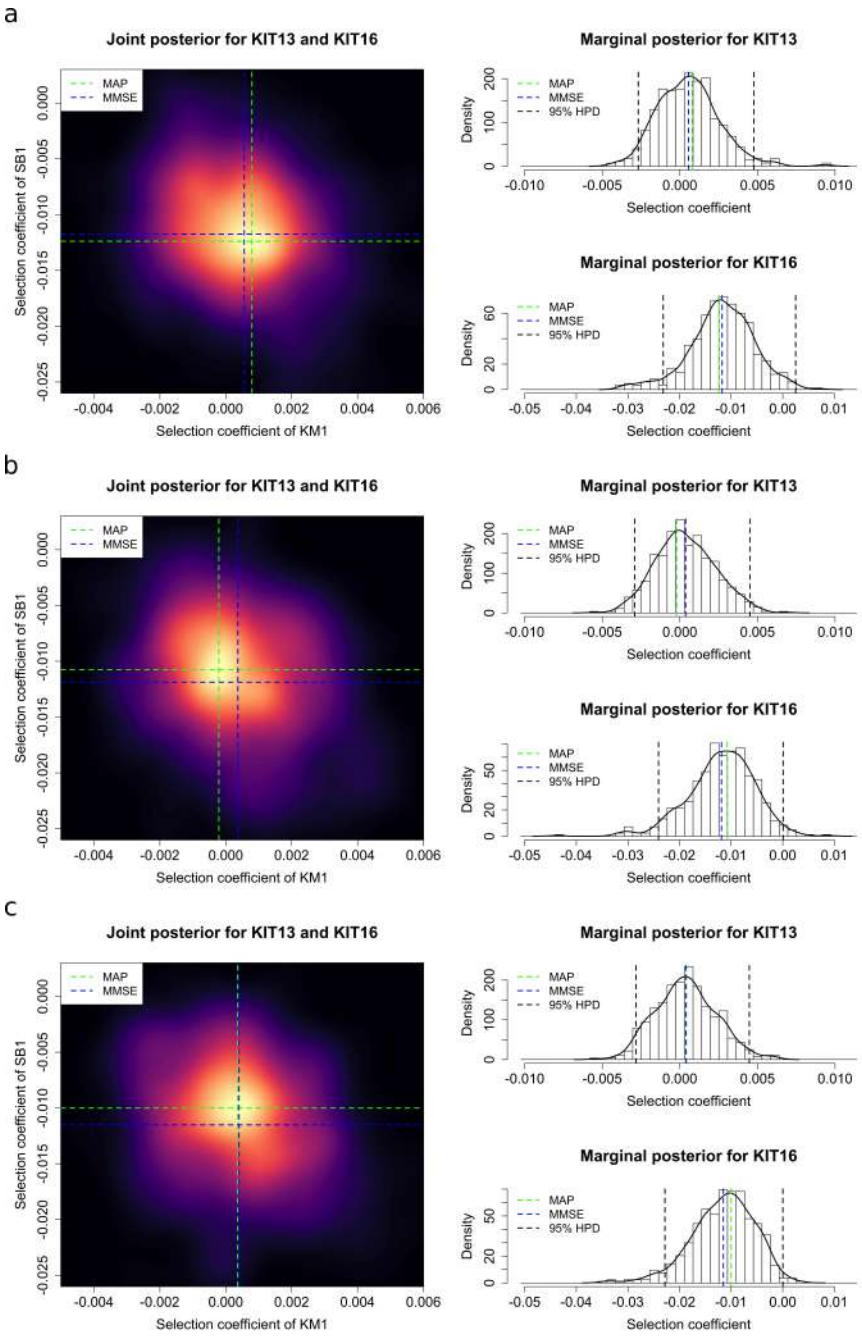

Figure 4.10: Posterior probability distributions for *KIT13* and *KIT16* obtained using the two-locus method with the population size of 16000 from the samples dated from 5472 years BP, with average rate of recombination (a) 5×10^{-9} crossovers/bp. (b) 1×10^{-8} crossovers/bp. (c) 5×10^{-8} crossovers/bp.

The estimates of the selection coefficients suggest that the *KM1* allele at the *KIT13* locus is weakly positively selected whereas the *SB1* allele at the *KIT16* locus is strongly negatively

	recombination rate	MAP ($\times 10^{-2}$)	MMSE ($\times 10^{-2}$)	95% HPD ($\times 10^{-2}$)
KIT13	0.234×10^{-4}	0.079	0.056	$[-0.268, 0.476]$
	0.469×10^{-4}	-0.021	0.037	$[-0.292, 0.451]$
	2.340×10^{-4}	0.036	0.040	$[-0.283, 0.447]$
KIT16	0.234×10^{-4}	-1.238	-1.175	$[-2.316, 0.025]$
	0.469×10^{-4}	-1.076	-1.187	$[-2.407, 0.007]$
	2.340×10^{-4}	-1.001	-1.152	$[-2.283, 0.002]$

Table 4.9: MAP and MMSE estimates, as well as the 95% HPD intervals, for *KIT13* and *KIT16* obtained by using the two-locus method with the population size of 16000 from the samples dated from 5472 years BP.

selected. The 95% HPD intervals for both selection coefficients include the value of 0. For the *KIT13* locus, the posterior probability for positive selection is 0.564, which is strong evidence for positive selection. In comparison, for the *KIT16* locus, the posterior probability for negative selection is 0.982, which can be regarded as a piece of strong evidence to support the *SB1* allele at the *KIT16* locus being negatively selected. To be more convinced, we use a set of different population sizes, $N = 8000$, $N = 16000$ and $N = 32000$, to re-run our two-locus method and there are no obvious changes in selection coefficients estimates for both mutant allele on *KIT13* locus and *KIT16* locus. The results for different population sizes are displayed in the Appendix.

In addition, we also estimate the selection coefficient using the single-locus method, which is the first step of the 'two-step' method described in Chapter 4. The estimate of the selection coefficients suggests that the *KM1* allele at the *KIT13* locus is weakly selectively advantageous which is the same as the result from our two-locus method. The *SB1* allele at the *KIT16* locus appears weakly selectively deleterious, using the single-locus method, with the posterior probability distributions for the *KIT16* locus is approximately symmetric about 0. This suggests that there is no evidence to support the *KM1* allele at the *KIT13* locus or the *SB1* allele at the *KIT16* locus being selected. Compared to the results from the two-locus method, the single-locus method fails to detect negative selection at the *KIT16* locus without modeling genetic recombination and local linkage.

	MAP ($\times 10^{-2}$)	MMSE ($\times 10^{-2}$)	95% HPD ($\times 10^{-2}$)
KIT13	0.006	0.005	$[-0.363, 0.362]$
KIT16	-0.023	-0.024	$[-0.713, 0.590]$

Table 4.10: MAP and MMSE estimates, as well as the 95% HPD intervals, for *KIT13* and *KIT16* obtained by using the single-locus method with the population size of 16000 from the samples dated from 5472 years BP.

Our finding from the two-locus method using this *KIT* data set is compatible withWutke et al. [128]. By analysing the fluctuation of the allele frequencies at consecutive sampling time points, Wutke et al. [128] generates a result that the spotted horses since the Middle Ages lost

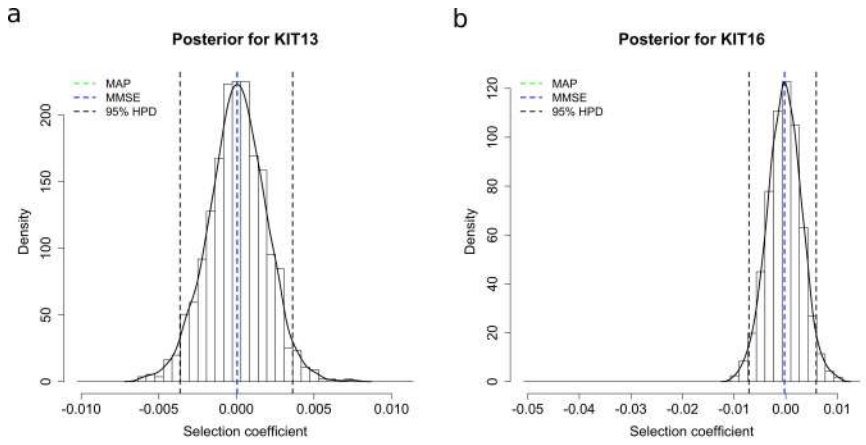

Figure 4.11: Posterior probability distributions for *KIT13* and *KIT16* obtained by using the single-locus method with the population size of 16000 from the samples dated from 5472 years BP. (a) *KIT13*. (b) *KIT16*.

attractiveness. They suggest some good explanation for decreasing of attractiveness in spotted horses, firstly, according to ancient Roman records, solid horses were preferred to spotted horses as the latter was considered to be of inferior quality. Besides, in Medieval religious culture, the spotted horses had a lower religious prestige which is regarded to have a negative connotation after several epidemics, especially after the Black Death. Additionally, the preference for spotted may have benefit from being able to distinguish visually domestic from wild horses at the early stage of domestication, and later, such requirement decreased with the decline of wild horse populations. Finally, as long-range weaponry developed, for example, longbow and ballista, the spotted pattern may result in the rider of the spotted horse to be more easily targeted than the solid horse by such long-range weaponry, especially over long distances or while moving

In this real data study, the single-locus method ignores genetic recombination and local linkage and fails to supply evidence of selection for mutant allele on both loci. However, the two-locus method achieves estimates for both selection coefficients and shows there is no strong evidence for the *KM1* allele at locus *KIT13* to be positively selected, but there is strong evidence for the *SB1* allele at locus *KIT16* to be negatively selected. This result supports the existing publication investigating the locus *KIT13* and locus *KIT16* which suggests there may exist a negative selection for the mutant allele on locus *KIT16* [128]. The two-locus method is more promising and desirable than the single-locus method in this real study. Furthermore, if we ignore the genetic recombination and local linkage to use the single-locus method, we can hardly detect a piece of obvious evidence for negative selection on locus *KIT16*. It suggests that when segregating alleles are tightly linked, for example, in this real data study, the locus distance between *KIT13* and *KIT16* is 4688 bp, considering the genetic recombination and local linkage is very necessary and the two-locus method is more faithful.

4.7 Discussion

In this work, we developed a novel MCMC-based method to infer natural selection at a pair of linked loci from allele frequency time series data while accounting for the effect of genetic recombination effect and local linkage information. The two-locus method is based on an HMM framework incorporating the two-locus Wright-Fisher diffusion with selection. One limitation of our approach is that it assumes that mutant alleles were created before the initial sampling time point. However in the case of earlier samples without the mutant allele, there is uncertainty in pinpointing when the mutant allele arose. Co-estimating selection coefficients at linked loci along with their allele ages can be expected to be cumbersome as there are many cases to consider. Another limitation is the two-locus method assumed the selection coefficients are constant via time, however, there are many recent publications pointing out that the selection coefficient may vary a lot, especially before and after domestication. It motivates us to make a further extension of the two-locus method to capture time-varying selection information and make inferences about that. The diffusion approximation of the Wright-Fisher model provides the potential to realize that, but the main problem is, when we introducing time-varying selection coefficients into the model, how to make an effective inference based on such high-dimensional model using limited information of time-series data.

In the case of the ancient horse data, we did not wish to make the assumption that the mutation arose earlier than the time of the first sample. However, we can compare the inference results obtained with different choices of initial sample time and reach the same conclusion that there is no strong evidence for the *KM1* allele at locus *KIT13* to be positively selected, but there is strong evidence for the *SB1* allele at locus *KIT16* to be negatively selected. In these cases, the dominate parameter can be evaluated by how the alleles dominate their phenotype since horses still live and usually been seen on our planet. If we need to applied our two-locus method to some extinct species dataset, we can regard the dominate parameters as parameters of genetic interest to jointly estimate with selection coefficients using an estimated effective population size. Besides that, *KIT13* and *KIT16* are tightly linked with small genetic distance and both of them control the pattern of leather. It results in the two-locus is more promising as in such circumstances they are more suitable to be considered together. However, there mush exists many cases that more loci should be considered as linked. It also motivates us to extend the existed method to multi-locus. Our Bayesian statistical framework has the potential to being extended to infer natural selection at multiple linked loci from time-series data of allele frequencies. At least, we can employ the composite-likelihood method to pairwise-jointly make inferences between multi-locus cases, those relevant work will be studied further and might improve the results of the natural selection inference.

CHAPTER 5

BAYESIAN INFERENCE OF DEMOGRAPHIC HISTORY FROM WHOLE-GENOME DATA

Thanks to advanced sequencing method, genome-wide sequencing data is increasingly available which is highly informative for inferring demographic history. Approximate Bayesian computation (ABC) has proved to be promising for uncovering the population structure and inferring population genetic parameters. However, whole-genome data poses challenges for the ABC method because it may be computationally infeasible to simulate long regions of the genome and compute many complex summary statistics using big data. In this chapter, I will present a method to make an estimate using the ABC method for a given local region of genome and employ the Expectation Propagation (EP) method to combine estimates of genetic parameters of interests from different sites together.

5.1 Introduction

Since approximate Bayesian computation was first introduced into population genetics [15, 91], the method has been applied to an increasing range of complex model-based inference problems. As I have discussed in Chapter 2, the ABC techniques that have been developed promise a good estimate of parameters [13, 14, 35, 117]. There are increasing numbers of ABC applications in population genetics: for example, Jay et al. [58] has used the ABC method to infer paleolithic and neolithic human expansion based on whole-genome sequence data, Raynal et al. [94] combine random forests and ABC method to construct a Bayesian inference method which is employed to uncover human population genetics given data from the 1000 Genomes Project Consortium, and Sheehan and Song [102] introduce a likelihood-free method based on ABC and deep learning, applying the method to analysis of 197 Drosophila melanogaster genomes

from Zambia to investigate the historical changes in the effective population size, and selective landscape. Beaumont [13] provided an excellent review of ABC's recent development and its applications.

However, the computation time of simulation-based methods like ABC is highly dependent on the size of the observations. Whole-genome sequencing data is very informative for uncovering the demographic history, but the relevant information is contained in large datasets. It is infeasible for the standard ABC method which is illustrated in Chapter 2 to simulate thousands of such big datasets and compute complex summary statistics on them. Compared to the use of whole-genome sequencing data, it is more manageable to make inferences on split genome data. Last decades, many publications turn to split-data likelihood approaches, for example, Rydén [97] propose a maximum split data likelihood estimate method based on the hidden Markov model. We can employ a composite likelihood assumption to find the factorizing likelihood by multiplying the likelihood from each split dataset, where the split dataset is also known as a component. To infer population genetic parameters, if we split data into many chunks of large size, in which the chunk is large enough to allow the effect of linkage to be ignored, then we can approximate the global likelihood by composite likelihood. Based on this assumption, we can use the Monte Carlo method or ABC method to compute the posteriors on each chunk and combine the information from each site together.

Expectation Propagation is introduced by Minka [84], in which they extend the assumed density filtering method to incorporate iterative refinement of the approximations and the EP method has proved to be one of the most popular methods in Bayesian machine learning [25]. Gelman et al. [43] summarise that the intuitive idea of EP is to work at each step with a "tilted distribution" that combines the likelihood for different sites of the data with the "cavity distribution", which is the approximate model for the prior and all other parts of the data. EP iteratively approximates the moments of the tilted distributions and incorporates those local approximations into a global posterior approximation. In this way, EP can be used to divide the computation for large models into manageable sizes. They also suggest that the moments of multivariate tilted distributions can be obtained by a variety of approximation methods, for example, MCMC, Laplace approximations, importance sampling and so on [43]. Li et al. [68] use of a method named stochastic EP which can maintain a global posterior approximation but update it locally. Stochastic EP aims at ducing the number of approximating factors that results from the increase in the number of the data points, which leads to a large memory overhead. Dehaene and Barthelmé [25] propose a similar variety of stochastic EP which they call average-EP and prove, in the limit of infinite data, the iterations of both averaged EP and EP are same: they behave like iterations of Newton's algorithm for finding the mode of a function [25]. Seeger and Nickisch [101] develop a method based on covariance decoupling techniques to solve the EP relaxation of Bayesian inference for continuous-variable graphical models. Teh et al. [114] propose a stochastic natural gradient expectation propagation method that does not require any

simplifying assumptions on the distribution of interest and has a better convergence performance than standard EP. Barthelmé and Chopin [7] introduce EP into the ABC algorithm and propose the EP-ABC method, which they find is faster by a few orders of magnitude than the standard ABC algorithm and supplies an EP structure to split data into many chunks in where we can use local summary statistics $||s_i(y_i) - s_i(y_i^{obs})|| \leq \epsilon$ instead of whole data summary statistics $||\mathbf{s}(\mathbf{y}) - \mathbf{s}(\mathbf{y}^{obs})|| \leq \epsilon$. It improves ABC by having more possibilities for accepting the simulated samples in each chunk which are then combined to approximate the posterior distribution for the full data. Instead of using the EP method, Scott et al. [100] present a consensus Monte Carlo method, which is highly related to the EP-ABC method. Consensus Monte Carlo method yields random draws from the sub-posterior with assumption that the Bernstein-von Mises theorem holds and the target distribution can be approximated by a multivariate Gaussian. In addition, consensus Monte Carlo method Scott et al. [100] proposes to fit multivariate Gaussian to the samples from each sub-posterior and multiply the densities together [13].

In this Chapter, I will start with how to apply the ABC method described in Chapter 2 to infer population genetics parameters. After that, I will introduce the concept of EP in detail and employ the EP method to develop an ABC based method inferring demographic parameters using a long-region genome sequencing dataset. Finally, I will present the real data analysis results from Bayesian inference employing EP.

5.2 ABC applications in Population Genetics data

For simplicity, let us consider an isolation-migration (IM) model with migration matrix denoted as $\mathscr{M}$. The element contained in migration matrix $\mathscr{M}$ denoted as $m_{i,j}$ where $m_{i,j}$ means migration rate from population i to population j. In addition, we denote N_j to be the effective population size for different population and $\theta_j = 4 \times N_j \times \mu$. We often regard the mutation rate μ is constant per base-pair per generation, and instead of inferring effective population size N_j, we can estimate θ_j and obtain the estimate of effective population size N_j by dividing by mutation rate μ. As there is only one divergence time in our IM model, we denote the time of the divergence as T. Figure 5.1 presents the isolation-migration model with parameters. The recombination rate is denoted as $\rho = 4 \times N_A \times r$ where N_A is the effective population size of population ancestral. The genetic parameter of interests denote as ϕ, in our IM model, it can be $\phi := \big(\theta_A, \theta_1, \theta_2, T, \rho, m_{1,2}, m_{2,1}\big)$.

The data we generated to investigate is haplotype data, which can be simulated by Hudson's *ms* simulator [55]. *ms* uses the Monte Carlo method to generate samples from the Wright-Fisher neutral model with assuming an infinite-sites model of mutation. It allows a variety of demographic histories and coding on language C. In addition, *msms*, containing all functions of *ms*, is a coalescent simulation program for a structured population with the selection at a single diploid locus [32]. *msprime* is another implementation of Hudson's *ms* algorithm using sparse trees and coalescence records as the key units of genealogical analysis. *msprime* has proved to be

faster than any other simulators when simulating a large number of samples [61]. A simulator that generates haplotype data using the sequential coalescent with recombination model is known as *scrm* [109], which can efficiently approximate the coalescent with recombination [79].

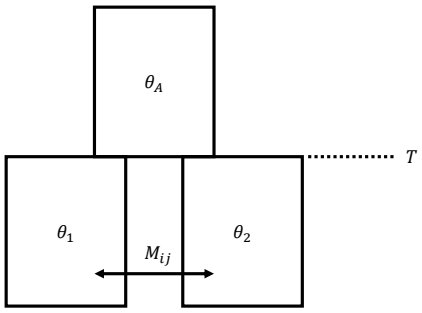

Figure 5.1: isolation-migration (IM) model, $\theta_j = 4 \times N_j \times \mu$ where μ is the mutation rate, N_j is the effective population size for different population, e.g., N_A is the effective population size for ancestral population, which is before divergence time. T is the time of the divergence. $m_{i,j}$ is the element contained migration matrix $\mathscr{M}$.

The syntax of these simulators is similar to Hudson's *ms*, and here I choose to use *scrm* to generate the haplotype data from the IM model for our examples to illustrate the calculation of summary statistics and the performance of ABC algorithm using genome data. In this example, the length of each haplotype is set to be 20kb and the total phenotypes draw from the two sub-population is 100 in which 40 phenotypes from sub-population 1 and the other 60 phenotypes from sub-population 2. I set the parameter $\theta_1 = 0.02$ corresponding to the present effective population size (N_1) and $\rho = 0.015$. At the time of the divergence $T = 200$, which is also scaled with $4 \times N_1$, the effective population size for the sub-population 2 is $5 \times N_1$. The effective population size of ancestral is $10 \times N_1$. In simulation data, or each position, it contains two possible values 0 or 1, representing ancestral state and derived state, respectively.

5.2.1 Summary statistics of population genetics data

In Chapter 2, I have discussed the importance of choosing summary statistics for the ABC algorithm. Although we can employ the semi-automatic ABC method [35] and local regression postponed adjustment method [15] to improve the performance of ABC, the summary statistics choice is often very subjective. Here I will show the summary statistics, which are employed to summarise all haplotype sequencing data in this chapter, are closely related to the genetic

concept, e.g., the total number of segregating sites, summarised site frequency spectrum, Tajima's D index and so on.

segregating sites refers to Watterson estimator, which describes the genetic diversity [120]. The $\theta = 4 \times N_e \mu$ can be estimated by Watterson estimator if assumptions that there is a sample of $n \ll N_e$ haploid individuals with infinitely many sites capable of varying are met, then the Watterson estimator of θ is $\hat{\theta_w} = \frac{S}{b_n}$ where the S is the number of segregating sites and $b_n = \sum_i^{n-1} \frac{1}{i}$.

π, which is mean pairwise difference across haplotype, is often used to estimate the degree of polymorphism within a population [86], which is $\pi = \sum_{i,j} = x_i x_j \pi_{i,j}$ where x_i is the frequency of i_{th} sequences and $\pi_{i,j}$ is is the number of nucleotide differences per nucleotide site between the i_{th} and j_{th} sequences. *Tajima's D*, which involves both measurements of *segregating sites* and π, is employed to identify sequences which fail to fit the neutral model at equilibrium between mutation and genetic drift [112].

$$D = \frac{\hat{k} - \frac{S}{\sum_i^{n-1} \frac{1}{i}}}{\sqrt{\left(\frac{\frac{n+1}{3(n-1)} - \frac{1}{\sum_i^{n-1} \frac{1}{i}}}{\sum_i^{n-1} \frac{1}{i}} \times S + \frac{\frac{2(n^2+n+3)}{9n(n-1)} - \frac{n+2}{n \sum_i^{n-1} \frac{1}{i}} + \frac{\sum_i^{n-1} \frac{1}{i^2}}{(\sum_i^{n-1} \frac{1}{i})^2}}{(\sum_i^{n-1} \frac{1}{i})^2 + \sum_i^{n-1} \frac{1}{i^2}} \times S(S-1) \right)}} \tag{5.1}$$

where S is the number of segregating sites, n is the number of samples and $\hat{k}$ is expected number of SNPs.

SFS refers to the site frequency spectrum, which is the distribution of the sampled allele frequency for a population and the shape of *SFS* is sensitive to demography history [31]. The allele frequency spectrum from a sample of n chromosomes is calculated by counting the number of sites with derived allele frequencies, for example,

$$L = \begin{pmatrix} 1 & 0 & 0 & 1 & 0 & 0 & 1 \\ 0 & 0 & 1 & 1 & 0 & 0 & 1 \\ 1 & 1 & 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 1 \\ 1 & 0 & 0 & 1 & 0 & 1 & 1 \end{pmatrix} \tag{5.2}$$

There are 7 SNPs and 5 samples in this example, the summation of derived allele frequencies for each SNP is $s = (3,2,2,5,1,1,4)$, so there are 5 spectrum which is $(1,2,3,4,5)$ with frequencies are $(2,2,1,1,1)$.

I also use the covariance matrix, *cov*, to reflect the linkage disequilibrium information of haplotype sequences. The square of the Pearson coefficient of correlation r^2 is a common method to calculate linkage disequilibrium[118], in which I calculate the covariance matrix of sample like L in Equation 5.2 and use the mean and standard deviation of the upper triangular elements in the covariance matrix as summary statistics. *Fay and Wu's H* is another statistic to identify

sequences that have experienced selective sweeps [33]. I also use the summation of a scaled pairwise difference for each sub-population, which is a good approximation to the value of $(1-F_{st})$.

5.2.2 Simulation study for one chunk haplotype sequences data

In this section, I will present a simulation study to illustrate how to use the ABC method with summary statistics introduced in the former section to make estimates of genetic parameters based on sequences data from IM model.

The total number of the summary statistics I calculate here is 70 and by using semi-automatic method, I project 70 summary statistics down to one for each parameter and use the projected summary statistics to infer the genetics parameter of interests ϕ. For simplicity, I reduce the parameter space to 3, which is $\phi = (\theta_1, \rho, T)$. $\theta_1 = 4 \times N_1 \times \mu$ where N_i is present effective population size for population i and μ is mutation rate. At the time of the divergence $T = 200$, which is also scaled with $4 \times N_1$, the effective population size for the sub-population 2 is $5 \times N_1$. The effective population size of ancestral population is $10 \times N_1$. The migration matrix is set to be fix where $m_{1,2} = 0.005$ and $m_{2,1} = 0.001$. The total number of sampled haplotypes is 100, which 40 is for population 1 and the rest 60 is for population 2. The total length of the haplotype sequences data is 20 kb. The parameters involved in this simulation are positive for θ_1, ρ, T, so here I choose the Normal distribution as a prior on the log-transformed parameters, which is

$$log\theta_1 \sim N(\mu_1, \sigma_1^2) \tag{5.3}$$

$$log\rho \sim N(\mu_2, \sigma_2^2) \tag{5.4}$$

$$logT \sim N(\mu_3, \sigma_3^2) \tag{5.5}$$

We denote $\mu = (\mu_1, \mu_2, \mu_3)$ and $\sigma = (\sigma_1, \sigma_2, \sigma_3)$. In the first example, I choose $\mu = (0.5, 0.5, 5)$ and $\sigma = (0.5, 0.5, 1)$. The total simulation number is 10^5 and the proportion of acceptance is 0.01. The true value is chosen to be $\phi = (2, 1.5, 200)$, and the marginal distribution of accepted ABC samples is shown in Figure 5.2.

As we can see, ABC using the projected 70 summary statistics perform well. The mean of the accepted ABC samples for θ_1 and T are very close to their true value. The ABC mean posterior of parameter ρ has a little bias. I also summarise the result in Table 5.2. If we use the mean of the marginal posterior distribution of accepted ABC samples as the point estimate of the parameter, the bias is moderately small for all parameters. However, a concern is that the ABC marginal posterior distribution of the parameter ρ shows a skewed distribution.

To investigate further, I have run 200 replicate simulation studies based on the same true parameters value $\phi = (2, 1.5, 200)$ and use the same set of summary statistics and prior distribution, $\mu = (0.5, 0.5, 5)$ and $\sigma = (0.5, 0.5, 1)$, to infer the genetic parameters of interest and store

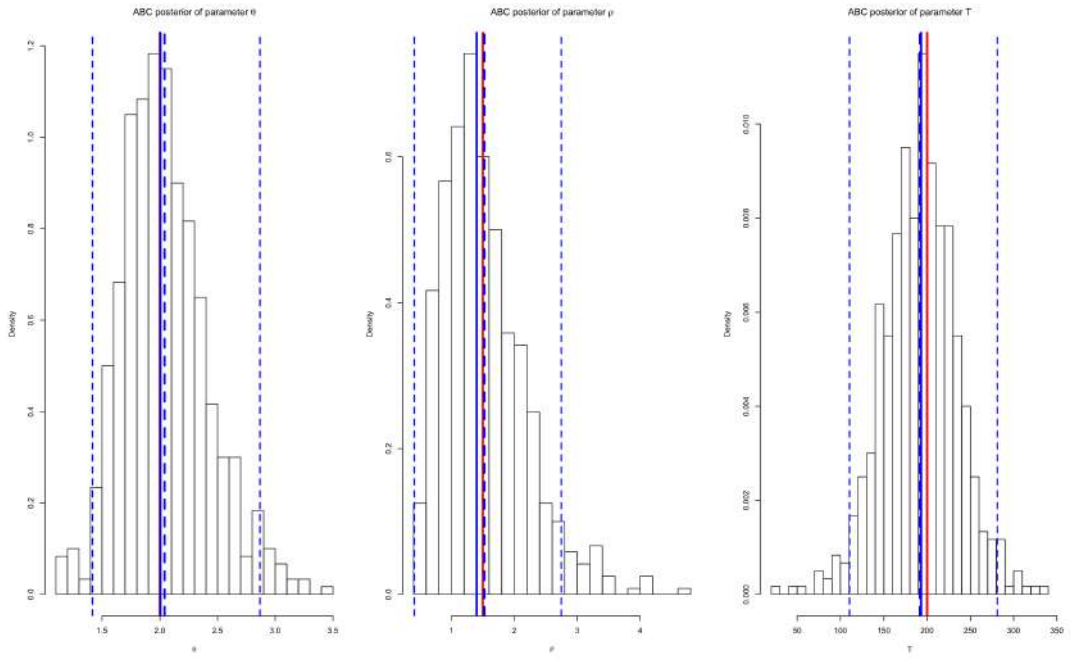

Figure 5.2: The marginal distribution of accepted ABC samples simulated from the IM model. The blue dashed vertical lines represent the boundaries of 95% highest posterior density interval. The medium blue dashed vertical line is the median of the accepted ABC samples and the blue solid vertical line is the mean of the accepted ABC samples. The red line represents the true value for each parameter. The histograms for parameters from left to right are θ_1, ρ, T.

Parameter with its true value	mean	median	95% HPD interval
$\theta_1 = 2$	2.041	2.005	[1.416, 2.867]
$\rho = 1.5$	1.523	1.402	[0.4110, 2.745]
$T = 200$	191.49	193.11	[110.438, 281.141]

Table 5.1: The summary of ABC marginal posterior distribution

the mean of the marginal posterior distribution of ABC samples as the point estimate for each parameter. The results are presented in box-plot 5.3. To make the parameters into the same magnitude, I divide the estimated value of parameter T by 100.

Parameters	Bias	RMSE	$Bias_{bot}$	$RMSE_{bot}$
$\theta_1 = 2$	0.0035	0.2945	0.00327	0.29447
$\rho = 1.5$	0.0528	0.3575	0.05169	0.3576
$T = 200$	5.6561	37.4083	5.52576	37.3778

Table 5.2: The summary of 200 replicates simulation study. The Bias is the average bias across all replicates, The RMSE represents root mean square error. The $Bias_{bot}$ and $RMSE_{bot}$ are bootstrap Bias and RMSE value with bootstrap step is 10^5

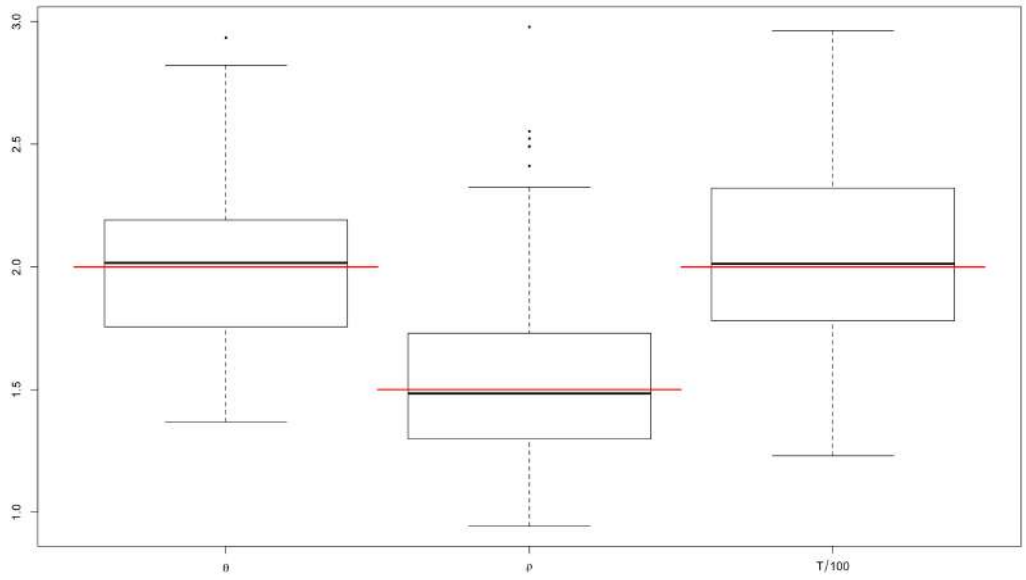

Figure 5.3: Box-plot of 100 replicates ABC simulation study. The tips of the whiskers denote the 2.5%-quantile and the 97.5%-quantile, and the boxes represent the first and third quartile with the median in the middle. The red solid line represents the true value for each parameter.

As we can see, the marginal posterior distribution of accepted ABC samples for parameter ρ is skewed and leads it to being more likely to deliver over-estimated value for the point estimate of ρ, which are the outliers above the upper whiskers. The average bias and RMSE for all parameters are relatively small with the scales of those true values. I also make a 10^5 bootstrap to achieve the bootstrap Bias and RMSE for each parameter, which have proved to be similar to our original calculation result for Bias and RMSE.

In addition, I have made another simulation study where I still set the prior distribution as $\mu = (0.5, 0.5, 5)$ and $\sigma = (0.5, 0.5, 1)$ but change the true value of genetic parameters of interests $\phi = (\theta_1, \rho, T)$ into a sequence of random value , i.e., $\phi = (\theta_1 \sim \mathscr{U}(0.5,3), \rho \sim \mathscr{U}(0.5,3), T \sim \mathscr{U}(100,300))$ where $\mathscr{U}(a,b)$ is Uniform distribution with lower boundary a and upper boundary b. For each parameter, I draw 200 random values from the Uniform distribution and use our ABC method with the same hyperparameters of the prior distribution. This example aims to show the hyper-parameters of the prior distribution is appropriate for estimating wide-scaled genetic parameters of interests. The result is presented in Figure 5.4.

In Figure 5.4 we can find that our ABC method can make a good estimate for the various value of θ_1 and T. The mean of the estimate for both θ_1 and T are close to the 45-degree line, which suggests the mean of them are close to their true value. In addition, nearly all of the HPD interval bars, i.e., the chocolate color bar in Figure 5.4, cross the 45-degree line, which implies the true value is contained in the 95% HPD interval for each trial of both parameter θ_1 and T.

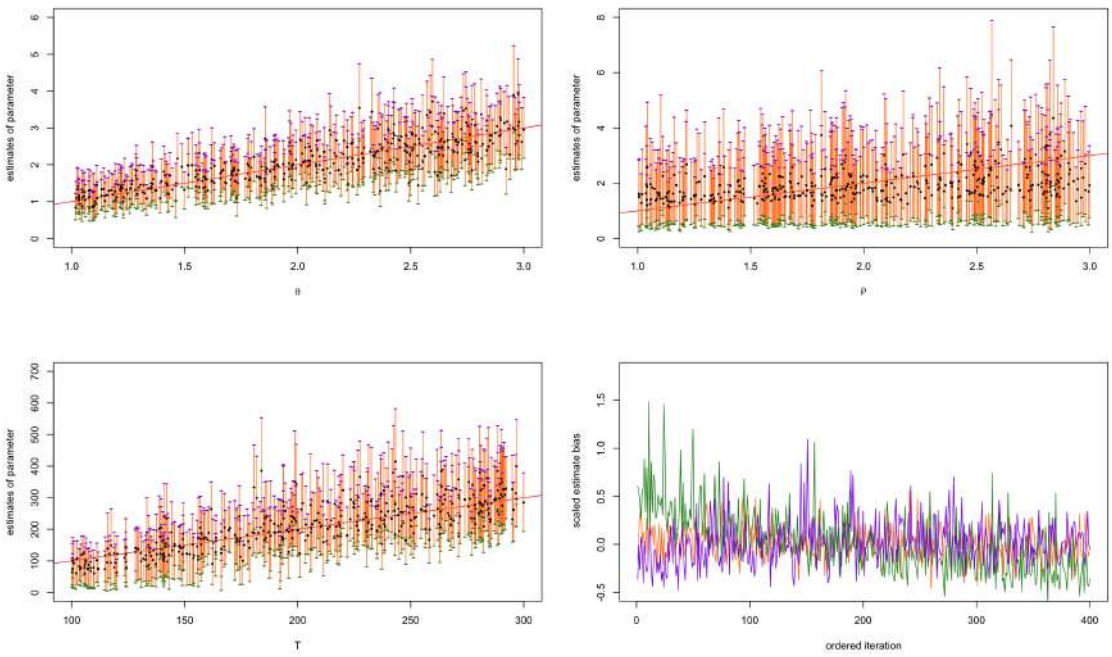

Figure 5.4: The figures index for top-left, top-right, bottom-left and bottom-right figure are 1,2,3,4, respectively. Figures 1-3 are the result for each replicates simulated data with the X-axis presenting the true value, which is used to generate the observed haplotype sequences, and Y-axis presenting the ABC estimates. The black point in the middle is the mean of the accepted ABC samples, the chocolate bar presents the 95% HPD interval with the upper boundary in color blue-violet and lower boundary in color forest-green. The red solid line is 45-degree line with intercept is zero. Figure 4 is the scaled estimates bias for (θ_1, ρ, T) with color chocolate, forest-green, blue-violet, respectively.

The scaled bias for i^{th} trial of k^{th} parameter in ϕ is

$$bias_{scaled} = \frac{\hat{\phi_i^k} - \phi_i^k}{\phi_i^k} \tag{5.6}$$

where the $\hat{\phi_i^k}$ is the point estimate from ABC, which is the mean value of the accepted ABC samples. As we can see, most of the trials for both θ_1 and T parameter have relative small bias and there are only a few fluctuations between ordered iteration 110 to 200. However, the scaled bias of parameter ρ implies that our ABC method does not perform quite well. Although nearly all of the true value ρ are contained in the 95% HPD intervals of our ABC estimates, the accepted ABC samples yield a large variation value of ρ, especially when ρ increases.

There are three potential explanations for the unexpected ABC performance on estimating ρ. Firstly, with 10^5 iterations of ABC, the acceptance proportion I used here is 0.01, which may not be enough small to reject the samples that should have been rejected. In addition, we can

find in Figure 5.2, the ABC posterior distribution of ρ tends to be skewed and has a relatively high frequency to have outliers in one-side. It leads to the issue that if we use an inappropriate prior distribution of ρ with relative large tolerance ϵ, the ABC estimate of ρ will be biased to one-side, as in Figure 5.4. In contrast, if we use an inappropriate prior distribution of ρ and small tolerance ϵ, it will lead to a very low acceptance rate. The last potential reason for this issue is that our summary statistics can capture information of parameter θ_1 and T well but can not extract information of parameter ρ from simulated data which seems most likely follows prior. In that case, our ABC method will perform badly regardless of the true value of ρ. The prior on ρ in this simulation study is $exp(N(0.5, 0.5))$, which has a mean 1.853 with standard deviation 0.982. The ABC performance for our trials, where the true value of ρ increase from 1.5 to 2.5, is much better than the trials for ρ less than 1.2 or bigger than 2.75. In conclusion, the ABC method based on the summary statistics I used here generally perform quite well, especial in estimating parameter θ_0 and T. The ABC estimate of parameter ρ is very sensitive to prior choice, which leads to motivation for us to find a method for refining the prior of such parameters.

5.3 Expectation Propagation updating with ABC weight

Whole-genome data is more frequently to be used to uncover the demographic history; however, such large scale genome data is computationally challenging for ABC. Recently, many publications propose the idea to use split data with Monte Carlo inferences instead of using whole data directly [100] [43]. We assume our full data $\mathbf{y}$ can be partitioned into L components $y_{1,\cdots,L}$, the posterior distribution with parameter ϕ can be written as,

$$p(\phi|\mathbf{y}) \propto \pi(\phi)p(\mathbf{y}|\phi) = \pi(\phi)\prod_{l=1}^{L} p(y_l|\phi) = \prod_{l=1}^{L} p(y_l|\phi)\pi(\phi)^{\frac{1}{L}} \tag{5.7}$$

Barthelmé and Chopin [7] propose an Expectation Propagation method involving a likelihood-free algorithm to obtain an EP approximation of the composite ABC posterior distribution, which can be written as,

$$\hat{p}_\epsilon(\phi|\mathbf{y}) \propto \prod_{l=1}^{L}\left(\pi(\phi)^{\frac{1}{L}}\int p(y_l|y_{1:l-1},\phi)\mathbb{K}_\epsilon(\parallel s_l(y_l) - s_l(y_l^{\text{obs}})\parallel)d\phi\right) \tag{5.8}$$

We can re-write Equation 5.8 as

$$\hat{p}(\phi|\mathbf{y}) \propto \prod_{j=1}^{L} g_j(\phi) \tag{5.9}$$

The intuitive idea of EP is to have an initial distribution of $g_j(\phi)$ and using sequencing of refinement based on local split data information to reshape the $g_j(\phi)$ distribution. There is a

figure in William Perry's dissertation [90] which is a good illustration of EP in Figure 5.5. It describes how an initial prior distribution, which is in yellow at the start point, reshape to a converged posterior distribution, which is the black curve at the endpoint. The prior distribution reshapes after the algorithm scanned the first site. Then the reshaped distribution move to the next site and repeat the former reshape procedures. After scanning the last site, the distribution is still reshaping, i.e., it does not converge. Then the reshaped distribution moves back to the first site and starts its new iteration scanning process until it converges where in this figure, the convergence point is the third iteration and the last site.

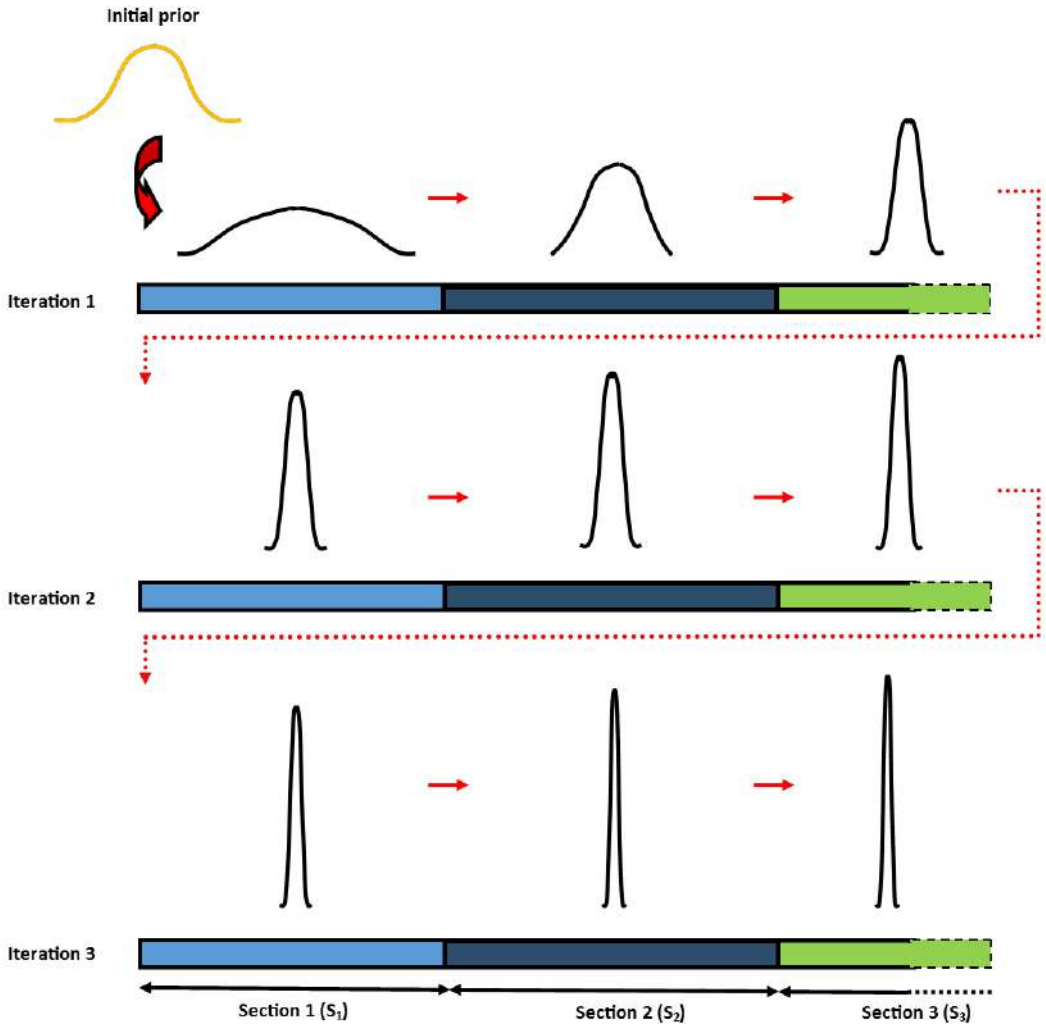

Figure 5.5: A figure from the dissertation written by William Perry [90] to illustrate how EP method sweep among the genome site by site.

As we can see, at the j^{th} step, the distribution $g_j(\phi)$ is refined with its local site data, here we

denote the cavity distribution as,

$$g_{-j}(\phi) = \left(\prod_{l=1}^{l=L} g_l(\phi)\right) \times \frac{1}{g_j(\phi)} \tag{5.10}$$

and tilted distribution of $g_j(\phi)$ as

$$g_j^t(\phi) \propto g_{-j}(\phi) p(y_j|\phi) \tag{5.11}$$

Then we can find a new value $g_j^*(\phi)$ by minimising the Kullback-Leibler divergence between distribution $g_{-j}(\phi)p(y_j|\phi)$ and $g_{-j}(\phi)g_j^*(\phi)$, where Kullback-Leibler divergence is,

$$KL(p \parallel q) = \int p(\theta) log\left(\frac{p(\theta)}{q\theta)}\right) d\theta \tag{5.12}$$

Barthelmé and Chopin [7] propose that if the $g_j(\phi)$ at each site is Gaussian, the minimising process between the tilted distribution with $g_{-j}(\phi)g_j^*(\phi)$ is equivalent to matching the moments of $g_{-j}(\phi)g_j^*(\phi)$ with the moments of $g_{-j}(\phi)g_j^*(\phi)$. After obtaining the moments of tilted distribution, i.e., ξ_t, we can use transform mapping $\eta(\cdot)$ to obtain the natural parameter of the tilted distribution, which is $\eta_t = \eta(\xi_t)$. Here I denote the natural parameter of global parameter as η, then the we can have the the natural parameter of cavity distribution at site j as $\eta_{-j} = \eta - \eta_j$ and update the natural parameter at site j by setting $\eta_j = \eta_t - \eta_{-j}$. After that, set the global natural parameter $\eta = \eta_t$ until the η convergence. Then we find the distribution of the parameter of interest by using inverse transform mapping $\eta^{-1}(\cdot)$ to have the moments of parameters, i.e., $\xi = \eta^{-1}(\eta)$.

Now the only problem left is to estimate the moment of tilted distribution $\xi_t = (\xi_t^{(1)}, \xi_t^{(2)})$, where $\xi_t^{(1)}$ and $\xi_t^{(2)}$ are first and second moments of tilted distribution respectively, where

$$\xi_t^{(1)} = \frac{1}{N_t} \int \phi g_{-j}(\phi) p(y_j|\phi) d\phi \tag{5.13}$$

$$\xi_t^{(2)} = \frac{1}{N_t} \int \phi\phi^T g_{-j}(\phi) p(y_j|\phi) d\phi \tag{5.14}$$

where the N_t is the normalise constant which is

$$N_t = \int \phi g_{-j}(\phi) p(y_j|\phi) d\phi \tag{5.15}$$

To calculate this ξ_t, it is straight-forward to turn to the Importance sampling (IS) method. IS here is to simulate samples with parameter draw from the tilted distribution and accepting the

importance samples with acceptance ratio, at the same time, calculate the importance weight which is an unbiased estimate to likelihood. Then we can calculate the moments by using the importance weights. In contrast, in ABC, a similar weight can be obtain by the acceptance kernel $\mathbb{K}_\epsilon(\| s_l(y_l) - s_l(y_l^{\text{obs}}) \|)$ which is described in Chapter 2. As we know the importance weights, we have an opportunity to reuse the simulated dataset for the former site. Controlling the effective sample size is appropriate for each site, we can reuse the simulated dataset many times and make our ABC method computational power within EP structure. Here I purpose an EP method with ABC Algorithm in Algorithm 6 which is a variant of EP-ABC method [7]. This EP method with ABC Algorithm is similar to the Algorithm used in William Perry's dissertation which was originally developed and implemented by Mark Beaumont [90].

Algorithm 6 Expectation Propagation with ABC algorithm

1. Initial: For each site j, set initial parameter η_j^0 and summary statistics $s_j(\cdot)$
2. Set global natural parameter $\eta^0 = \sum_{j=1}^{L} \eta_j^0$, $\xi = \eta^{-1}(\eta^0)$, $\mathbb{K}_\epsilon(\| \cdot \|)$ and ϵ
3. For iteration $i = 1, 2, \cdots$,
4. For site $j = 1, \cdots, L$

$\eta_{-j} = \eta - \eta_j$ and $\xi_{-j} = \eta^{-1}(\eta)$

4.1 Sample $\phi_j \sim g_{-j}(\phi|\xi_{-j})$ and simulate $y_{sim} \sim \phi_j$

4.2 Project $s_j(y^{sim})$ to $s_j^{'}(y^{sim})$ with dimension N, where in $\phi^{[n]}$, $n = 1, \cdots, N$

4.3 Calculate

$$w_j^{[n]} = \mathbb{K}_\epsilon(\| s_j^{'}(y_j^{sim}) - s_j^{'}(y_j^{\text{obs}}) \|) \times \frac{N(\phi^{[n]}; \xi_j)}{N(\phi^{[n]}; \xi_j^{old})} \tag{5.16}$$

and

$$ESS = \frac{\left(\sum_{n=1}^{n=N} w_j^{[n]}\right)^2}{\sum_{n=1}^{n=N} (w_j^{[n]})^2} \tag{5.17}$$

If ESS $\leq ESS_{min}$, start from 4.1.

4.4 Compute estimates of moments for tilted distribution $\xi_t = (\xi_t^{(1)}, \xi_t^{(2)})$ by

$$\xi_t^{(1)} = \frac{1}{N_t} \sum_{n=1}^{N} w_j^{[n]} \phi^{[n]} \tag{5.18}$$

and

$$\xi_t^{(2)} = \frac{1}{N_t} \sum_{n=1}^{N} w_j^{[n]} \phi^{[n]} (\phi^{[n]})^T - \xi_t^{(1)} (\xi_t^{(1)})^T \tag{5.19}$$

where

$$N_t = \frac{1}{N} \sum_{n=1}^{N} w_j^{[n]} \tag{5.20}$$

4.5 Transform $\eta_t = \eta(\xi_t)$, set global $\eta = \eta_t$ and $\eta_j = \eta_t - \eta_{-j}$ for site j

If $ESS > ESS_{min}$, $j + 1 < L$, then $j = j + 1$ and start from 4.3

If $ESS > ESS_{min}$, $j + 1 = L$, then $j = 1, i = i + 1$ and start from 4.3

5. Stop when η converge
6. Return moment of global parameter $\hat{\xi} = \eta^{-1}(\eta)$ where $\hat{\phi} \sim N(\hat{\xi^{(1)}}, \hat{\xi^{(2)}})$

5.3.1 Simulation study for decomposited genome data

In this subsection, I will present the simulation study using the described in Algorithm 6. To be compatible with former simulation study and example in Section 2, here I use 70 summary statistics computed from the samples and employ the regression method to project those summary statistics to the same dimension with the genetic parameter of interests. The difference is that I use a bigger size of genome data, which is decomposed into 10 chunks. The total length of observed genome data is 200kb, so the chunk size for each site is 20kb. The population structure is the same as the former example. As we require the exponential family assumption for EP method, here I use multinomial Gaussian distribution for parameter $\phi = (\theta_1, \rho, T)$, for simplicity, I make a log-transform on each parameter here, which denote $\phi = (log(\theta_1), log(\rho), log(T))$. Then we have $\phi \sim N(\xi^{(1)}, \xi^{(2)})$, where mean vector $\xi^{(1)}$ and covariance matrix $\xi^{(2)}$ are

$$\xi^{(1)} = (\mu_1, \mu_2, \mu_3) \tag{5.21}$$

$$\xi^{(2)} = \begin{pmatrix} \sigma_{1,1} & \sigma_{1,2} & \sigma_{1,3} \\ \sigma_{2,1} & \sigma_{2,2} & \sigma_{2,3} \\ \sigma_{3,1} & \sigma_{3,2} & \sigma_{3,3} \end{pmatrix} \tag{5.22}$$

Here we choose a true value for ϕ to generate the observation for our simulation study with 200kb length. The migration matrix and effective population size changes are the same as the first example in Section 5.2.2. The total number of sampled haplotypes is 100, of which 40 is for population 1 and the rest 60 is for population 2. The length of the simulation study for ABC is 20kb. The number of simulation data generated for each chunk is 2×10^5 with the proportion of acceptance rate is 0.001. The number of iteration to sweep among the whole genome data is set to be 100. There are 40 replicates I run for this simulation study and 36 replicates have convergent results within 100 iterations. Then I use the outputs from 36 convergent replicates to generate the simulation study result. The true values for ϕ are shown in Equation 5.23.

$$\phi = (-1.2, -0.2, 3.5) \tag{5.23}$$

The starting value for the EP is

$$\xi_0^{(1)} = (2, 1.5, 5) \tag{5.24}$$

$$\xi_0^{(2)} = \begin{pmatrix} \frac{1}{2} & 0 & 0 \\ 0 & \frac{1}{2} & 0 \\ 0 & 0 & \frac{1}{3} \end{pmatrix} \tag{5.25}$$

Then we need to choose a prior distribution of ϕ to start our EP method, which here I set

$$\pi(\xi^{(1)}) = (3,3,5) \tag{5.26}$$

$$\pi(\xi^{(2)}) = \begin{pmatrix} 2 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 3 \end{pmatrix} \tag{5.27}$$

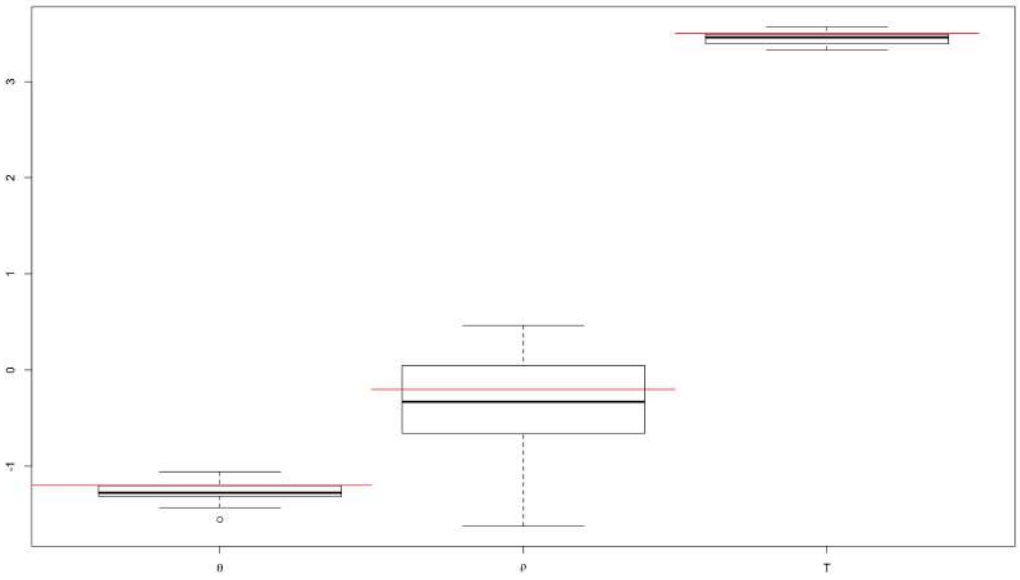

Figure 5.6: Box-plot of 36 replicates of simulation study. The tips of the whiskers denote the 2.5%-quantile and the 97.5%-quantile, and the boxes represent the first and third quartile with the median in the middle. The red solid line represents the true value for each parameter.

The result is presented in Figure 5.7. I take the mean value of different 36 replicates result, the estimated genetics parameters of interests are as below. The result is summarised in box-plot 5.6 and table 5.3.

$$\hat{\xi}^{(1)} = (-1.2638, -0.3415, 3.4459) \tag{5.28}$$

$$\hat{\xi}^{(2)} = \begin{pmatrix} 0.0408 & -0.0516 & 0.0106 \\ -0.0516 & 1.2199 & -0.0515 \\ 0.0106 & -0.0515 & 0.0246 \end{pmatrix} \tag{5.29}$$

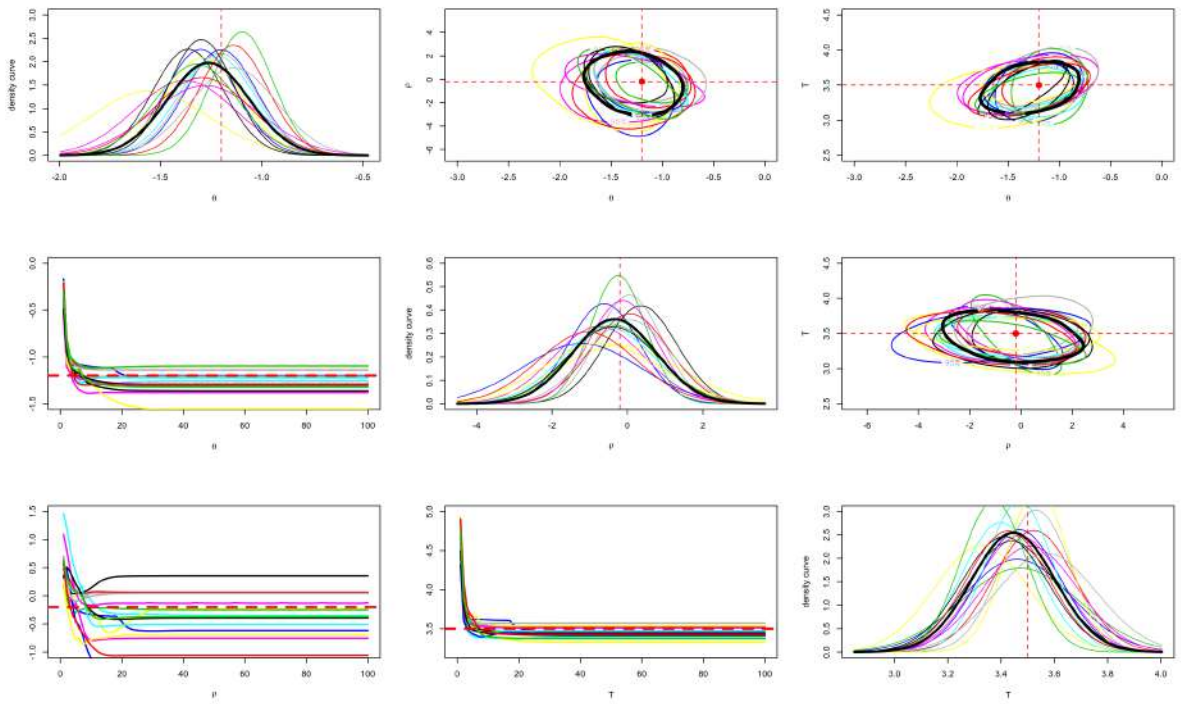

Figure 5.7: The simulation result from Algorithm 6. The black colour represents the mean value of all replicates results. The other different colours represents different replicate trajectory results from Algorithm 6. The Red dashed lines present the true values. The diagonal gives the marginal density curve for ϕ, the above diagonal set of curves shows the joint distribution between parameters of ϕ and the below diagonal figures indicate the convergence results for each element in the mean vector of ϕ.

Parameters	Bias	RMSE	$Bias_{bot}$	$RMSE_{bot}$
$\phi_1 = log(\theta_0) = -1.2$	-0.0638	0.1196	-0.0640	0.1192
$\phi_2 = log(\rho) = -0.2$	-0.1415	0.5103	-0.1441	0.5127
$\phi_3 = log(T) = 3.5$	-0.0541	0.08302	-0.0540	0.08303

Table 5.3: The summary of 36 replicates of simulation study. The Bias is the average bias across all replicates, The RMSE represents root mean square error. The $Bias_{bot}$ and $RMSE_{bot}$ are bootstrap Bias and RMSE value with bootstrap step is 10^5

The EP method performs well for this trial, comparing with former example, which we directly employed ABC method to infer parameter, the estimates of $\phi_1 = log(\theta_1)$ and $\phi_3 = log(T)$ are very more close to their true value. In box-plot 5.6 and table 5.3, both estimates of ϕ_1 and ϕ_3 yield very small variation. The estimate of $\phi_2 = log(\rho)$ has a little negative bias and has a bigger RMSE than the other two parameters, but it is still of sensible magnitude, which implies our EP method can deliver effective parameter estimates for all parameters involved in IM model. The total

length of haplotype sequence data is 200kb, and it is very time consuming to generate enough simulated dataset for such a long region to achieve effective ABC estimates for genetic parameters of interest. By using the EP method, we only need to generate a simulated dataset with a length of total length divided by the number of chunks. In addition, by using Algorithm 6, we can reuse the reference tables with the appropriate effective sample size, which allows us to generate simulated datasets infrequently. Also, the calculation of some summary statistics needs quadratic memory of the length of the simulated dataset. Using the EP method to make estimates site by site with a relatively small length of the simulated dataset can reduce the memory requirement to make inferences. Finally, the EP method with ABC algorithm only involves data from one site each time, i.e., in Algorithm 6-4.3, $\mathbb{K}_\epsilon(\| s_j^{'}(y_j^{sim}) - s_j^{'}(y_j^{\text{obs}}) \|$. Thus it does not suffer from a curse of dimensionality concerning the total number of chunks L and allows us to have a reasonable small tolerance to run our ABC algorithm. Barthelmé et al. [8] proposes a parallel variate of the EP-ABC method which can achieve 100-folder faster than standard ABC. In my simulation study, the EP method speeds up at least 15 times the ABC method I used in Chapter 2. The simulation study here only has 10 chunks, which leads to the comparison of computational power between EP-ABC and standard approach not being as obvious as it in Barthelmé et al. [8]. In conclusion, employing the EP method with ABC algorithm has more computational power to make estimates giving big data and supplies a good way to dealing with long region genome data which is often unmanageable for standard ABC method.

However, the EP method is sometime unstable and can not guarantee convergent result which is suffered from the Monte Carlo noise [49]. In my trial, I run 40 replicates and collect 36 replicates with converge output within 100 iterations. The convergence of mean vector often starts from around 20 iterations as in Figure 5.7, I also present the trajectories of elements in the covariance matrix in Appendix to show the convergence performance of this simulation study, as we can see that the majority of trajectories are convergent and some of them are still fluctuating in the last few iterations. Hasenclever et al. [49] propose a stochastic natural gradient expectation propagation (SNEP) as an alternative to EP method, in contrast to EP, this SNEP algorithm employs a double-loop structure to make sure it can deliver convergent results even faced with Monte Carlo noise in a complex model. For the further study, I am keen to investigate a method combining SNEP with likelihood-free Algorithm.

5.4 Real data study

In this section, I will present the analysis result from employing the EP method with ABC Algorithm 6 to uncover the demographic history of east African cichlids using whole genome data. The single nucleotide polymorphism (SNP) data is collected and previous analysed by Malinsky et al. [75]. There are detail data introduction about the whole genome data processing in the supplement document of Malinsky et al. [75]. To analysis this whole genome data, I

have constructed an IM model as in Figure 5.1, with the two sub-population 1 and 2 denote the population of Lake Massoko, which is a wider lake catchment area and the population of Astatotilapia calliptera (A. calliptera), which is crater lakes with high radiations. The data is published on DRYAD with total haplotypes are 170, in which 138 are from the Lake Massoko and 32 are from the A. calliptera. I choose the size of chunk is 200kb and the whole genome data of cichlid Scaffold 0 can be partitioned into 94 chunks. The maximum number of segregate sites for each chunk is round 1900.

The genetic parameters of interests are $\theta := \left(\theta_A, \theta_1, \theta_2, \rho, m_{1,2}, m_{2,1}, T\right)$. Since the majority of parameters are positive, here I use a logarithm to transform the parameters and denote $\phi := \left(log(\theta_A), log(\theta_1), log(\theta_2), log(\rho), log(m_{1,2}), log(m_{2,1}), log(T\mu)\right)$ where $\theta_k = 4 \times N_k \times \mu$ where μ is the mutation rate and $m_{i,j}$ represents the migration rate from population i to population j. The chunks size is 200kb and the iteration number is 200. For each chunk, the number of simulations to generate the reference table for ABC algorithim is 2×10^5 with an acceptance rate of 0.001. The summary statistics used in the analysis are the same as in the simulation study with 70 summary statistics in total and the projection method is also based semi-automatic regression method to project 70 summary statistics into the same number of parameter of interests, i.e., 7 projected summary statistics are used here. The prior distribution is set to be,

$$\pi(\xi^{(1)}) = (2,2,2,1,0,0,2) \tag{5.30}$$

$$\pi(\xi^{(2)}) = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & \frac{1}{2} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & \frac{1}{3} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & \frac{1}{3} & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix} \tag{5.31}$$

Here I run 5 replicates with same prior distribution and EP method with ABC Algorithm parameters setting, there are 2 replicates are not converge within 200 iterations. All 5 replicates trajectories are presented in Figure 5.8 to 5.14. The estimates for genetic parameters of interests are taken as the mean of the estimated values among 3 converge replicates, which is

$$\hat{\xi}^{(1)} = (3.2864, 5.0725, 5.7093, 3.3111, -0.5592, -1.2619, 3.9146) \tag{5.32}$$

$$\hat{\xi}^{(2)} = \begin{pmatrix} 0.0365 & -0.0113 & -0.0017 & 0.0084 & -0.0088 & 0.0355 & 0.0066 \\ -0.0113 & 0.0210 & -0.0054 & -0.0073 & 0.0057 & -0.0184 & 0.0176 \\ -0.0017 & -0.0054 & 0.0218 & 0.0046 & 0.0084 & 0.0114 & 0.0313 \\ 0.0084 & -0.0073 & 0.0046 & 0.0406 & -0.0053 & 0.0231 & 0.0029 \\ -0.0088 & 0.0057 & 0.0084 & -0.0053 & 0.1356 & -0.0211 & 0.0476 \\ 0.0355 & -0.0184 & 0.0114 & 0.0231 & -0.0211 & 0.1978 & -0.0345 \\ 0.0066 & 0.0176 & 0.0313 & 0.0029 & 0.0476 & -0.0345 & 0.4861 \end{pmatrix} \tag{5.33}$$

Here we use the average mean of posterior distribution of ϕ to estimate the genetics parameters of interests choose the mutation rate $\mu = 1.5 \times 10^{-8}$ per base pair to calculate the effective population size, recombination rate and the divergence time. The generation time is assumed to be 3 years per generation, all the genetic information used to calculate estimates of parameters were collected by Malinsky et al. [75].

$$N_1 = \frac{exp(\phi^{[1]})}{4 \times \mu \times 2 \times 10^5} = 2228.867 \tag{5.34}$$

$$N_2 = \frac{exp(\phi^{[2]})}{4 \times \mu \times 2 \times 10^5} = 13297.73 \tag{5.35}$$

$$N_A = \frac{exp(\phi^{[3]})}{4 \times \mu \times 2 \times 10^5} = 25138.32 \tag{5.36}$$

$$r = exp(\phi^{[4]} - \phi^{[1]}) \times \mu = 1.463 \times 10^{-8} \tag{5.37}$$

$$M_{1,2} = exp(\phi^{[5]}) = 0.5716662 \tag{5.38}$$

$$M_{2,1} = exp(\phi^{[6]}) = 0.2831156 \tag{5.39}$$

$$T = \frac{exp(\phi^{[7]})}{4 \times \mu \times N_1} \times 3 = 188487.1 \tag{5.40}$$

My findings are very similar to the EP-ABC analysis results presented in William Perry's dissertation using the algorithm is implemented by Mark Beaumont [90]. In his dissertation, he find his estimate of recombination rate is 3.75 times higher than the rate used in Malinsky

et al. [75], however, my analysis result does not have strong evidence to support that there exists a higher recombination rate than the previous study. The result shows evidence that the recombination rate should be the same as magnitude with the mutation rate. Besides that, the effective population size for sub-population 1, which is for Lake Massoko, is much smaller than the effective population size for A. calliptera. The migration rate from population Lake Massoko to population A. calliptera is $M_{2,1}$, which is half of the migration rate from population A. calliptera to population Lake Massoko, such skewed migration has also been detected by previous study [90]. In contrast with Perry and Beaumont [90], I find a little later divergence time which is around 188487 years ago, with the previous study is around 204765 years ago, but it is of a similar magnitude.

In conclusion, by using the method in Algorithm 6, we can have a very similar result with the previous study, which suggests the method I proposed here can deliver effective estimates to uncover demographic history using whole-genome data for this real data study. To compare with the previous Algorithm described in Perry and Beaumont [90], we have similar computational power, I employ 8 cores to parallel compute the estimates of the parameter with total time consuming is 39 hours per cores. The main difference is I calculate the projected summary statistics each simulation time and the proportion acceptance rate is half of the previous study, which means I need more computation to calculate the new projection matrix each time and to run more iteration for to obtain convergence. The results for each element in ϕ are presented in Figure 5.8 to 5.14 below.

5.5 Discussion

The EP method provides a structure that we can use to make estimates based on subdivided data, which is more manageable for the ABC algorithm. Under the appropriate assumption, we can factor those estimates from each site and target the global. In this chapter, I propose an EP method with an ABC algorithm that can make estimates using whole-genome data. I illustrate how to achieve an ABC estimate for each site in the second section and present an algorithm to apply the EP method in the third section with simulation study. By using that method, I deliver a re-analysis of results for whole-genome data from cichlids which is very similar to the previous study.

However, the drawback of the EP method as I have discussed that it can not make a guaranteed to converge. Vehtari et al. [119] points out the EP update step the KL divergence from the new global approximation to the tilted distribution $KL(g_{-j}(j)||g(j))$ is minimized by matching the moments. With a simulation-based method, the expectation of the new global approximation moments should then match with the tilted distribution moments. When working with the normal approximation, we would use the unbiased estimates of the mean and covariance of the tilted distribution, which are easily obtained from the simulated sample. However, using this

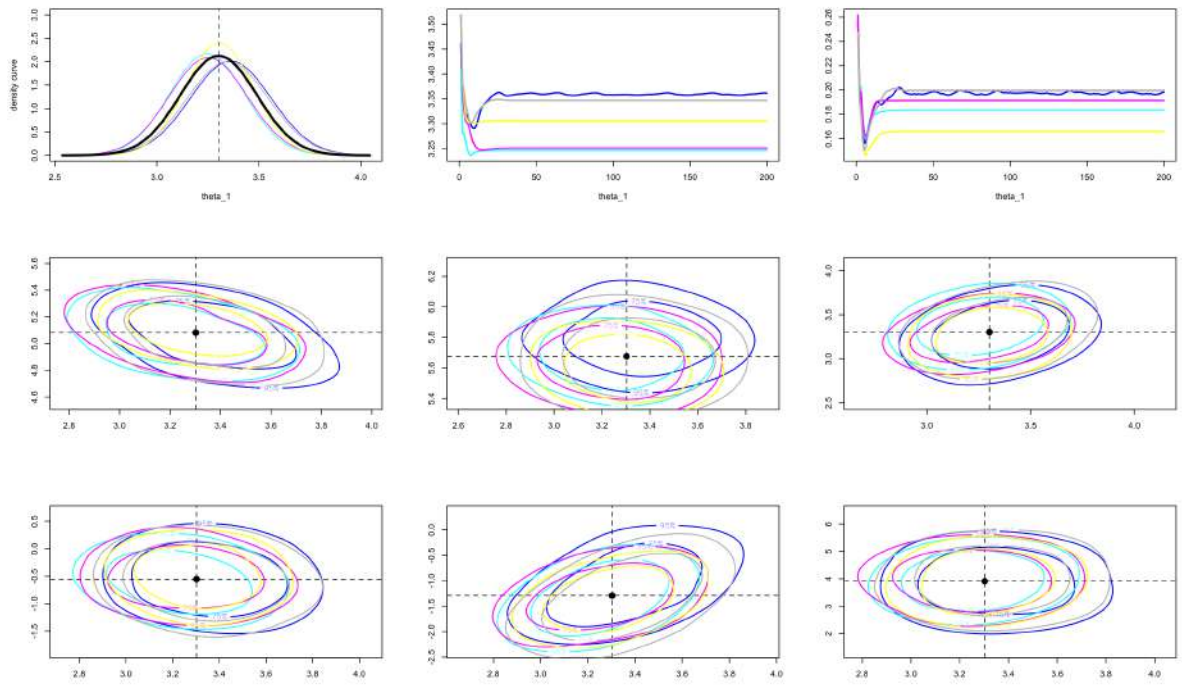

Figure 5.8: The EP method ABC Algorithm result for parameter $\phi^{[1]}$. The black curve and points present the mean of estimates from all 5 replicates. The coloured lines represent estimates from different replicates. The first plot in the top row is the marginal density curve of $\phi^{[1]}$ and the black curve represents the average estimates of parameter. The second plot in top row is the convergent performance for mean of parameter $\phi^{[1]}$. The remaining plots shows the 95% and 75% HPD curve between $\phi^{[1]}$ with $\phi^{[i]}$. If $i = 1$, then plot the convergent performance for the standard deviation of parameter $\phi^{[1]}$

estimator would not result in the least possible expected KL divergence in general. By solving this problem, Gelman et al. [43] has provided a distributed algorithm called stochastic natural gradient expectation propagation which has managed to be applied to the MCMC site inferences. In a further study, it may be a good point to apply the stochastic natural gradient expectation propagation with the EP-ABC method described in this chapter to overcome the convergence issue.

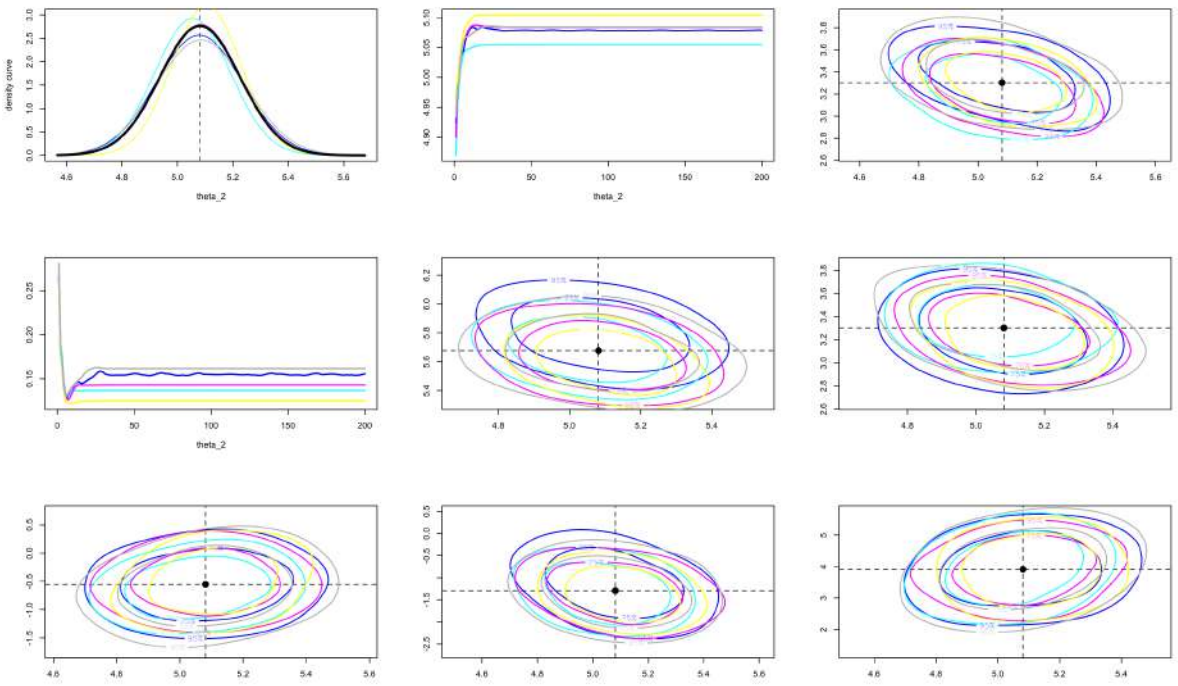

Figure 5.9: The EP method ABC Algorithm result for parameter $\phi^{[2]}$. The black curve and points present the mean of estimates from all 5 replicates. The coloured lines represent estimates from different replicates. The first plot in the top row is the marginal density curve of $\phi^{[2]}$ and the black curve represents the average estimates of parameter. The second plot in top row is the convergent performance for mean of parameter $\phi^{[2]}$. The remaining plots shows the 95% and 75% HPD curve between $\phi^{[2]}$ with $\phi^{[i]}$. If $i = 2$, then plot the convergent performance for the standard deviation of parameter $\phi^{[2]}$

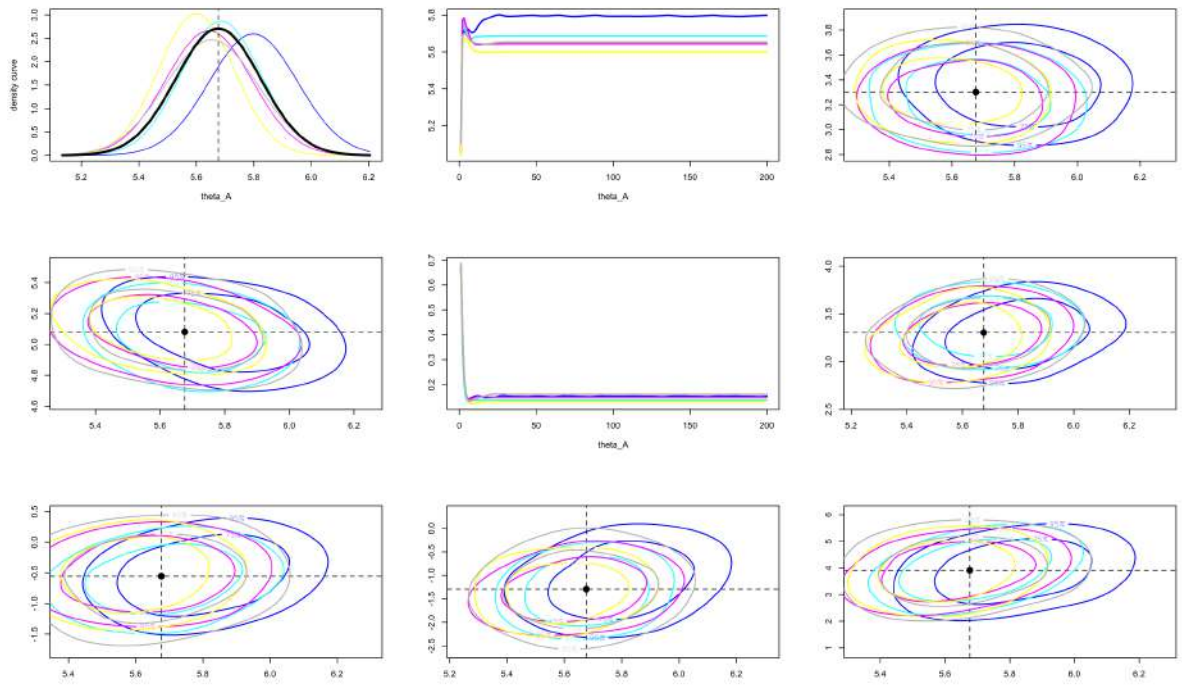

Figure 5.10: The EP method ABC Algorithm result for parameter $\phi^{[3]}$. The black curve and points present the mean of estimates from all 5 replicates. The coloured lines represent estimates from different replicates. The first plot in the top row is the marginal density curve of $\phi^{[3]}$ and the black curve represents the average estimates of parameter. The second plot in top row is the convergent performance for mean of parameter $\phi^{[3]}$. The remaining plots shows the 95% and 75% HPD curve between $\phi^{[3]}$ with $\phi^{[i]}$. If $i = 3$, then plot the convergent performance for the standard deviation of parameter $\phi^{[3]}$

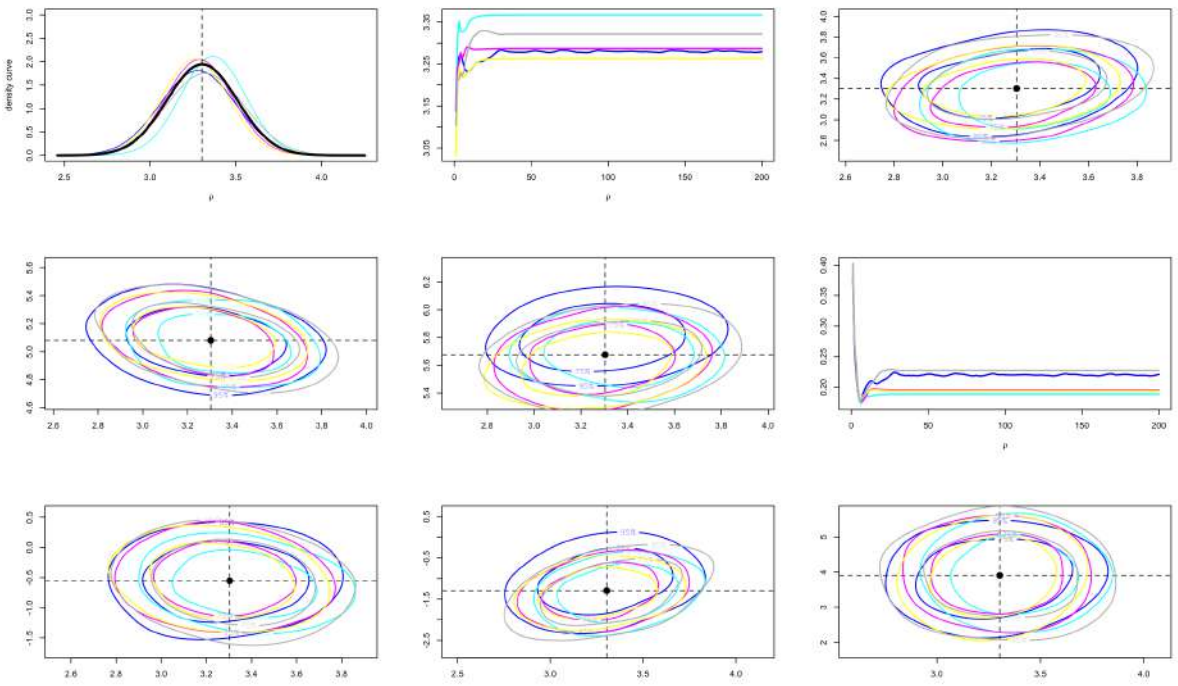

Figure 5.11: The EP method ABC Algorithm result for parameter $\phi^{[4]}$. The black curve and points present the mean of estimates from all 5 replicates. The coloured lines represent estimates from different replicates. The first plot in the top row is the marginal density curve of $\phi^{[4]}$ and the black curve represents the average estimates of parameter. The second plot in top row is the convergent performance for mean of parameter $\phi^{[4]}$. The remaining plots shows the 95% and 75% HPD curve between $\phi^{[4]}$ with $\phi^{[i]}$. If $i = 4$, then plot the convergent performance for the standard deviation of parameter $\phi^{[4]}$

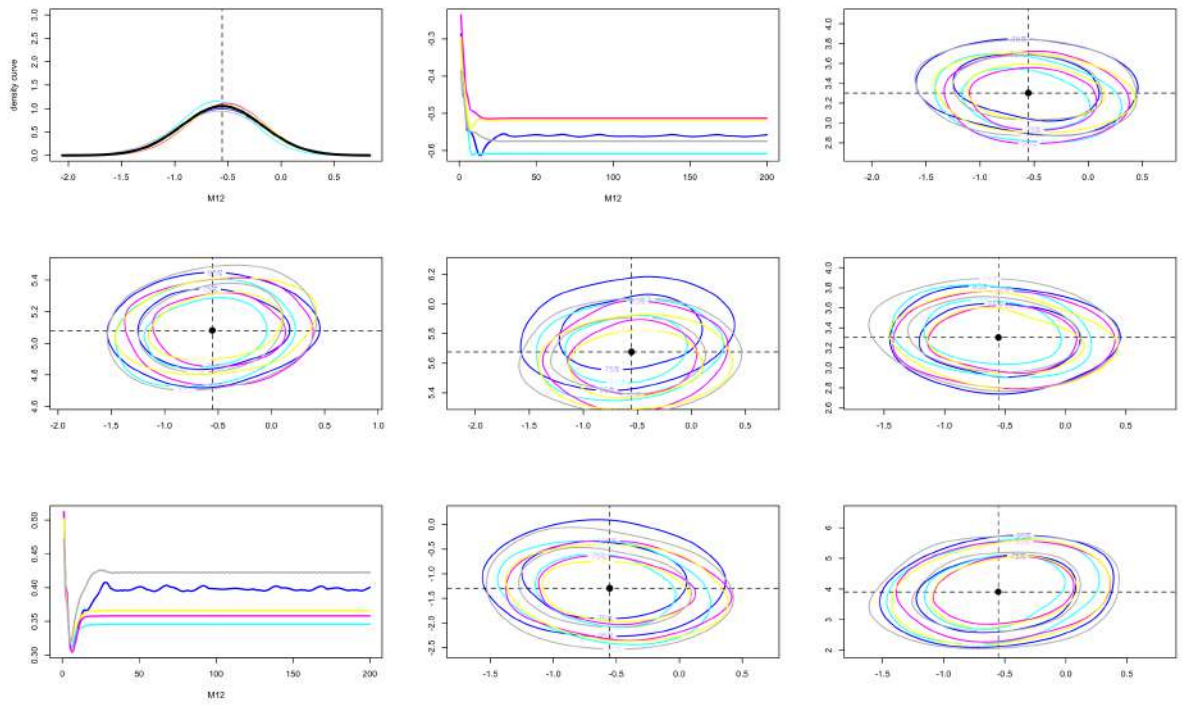

Figure 5.12: The EP method ABC Algorithm result for parameter $\phi^{[5]}$. The black curve and points present the mean of estimates from all 5 replicates. The coloured lines represent estimates from different replicates. The first plot in the top row is the marginal density curve of $\phi^{[5]}$ and the black curve represents the average estimates of parameter. The second plot in top row is the convergent performance for mean of parameter $\phi^{[5]}$. The remaining plots shows the 95% and 75% HPD curve between $\phi^{[5]}$ with $\phi^{[i]}$. If $i = 5$, then plot the convergent performance for the standard deviation of parameter $\phi^{[5]}$

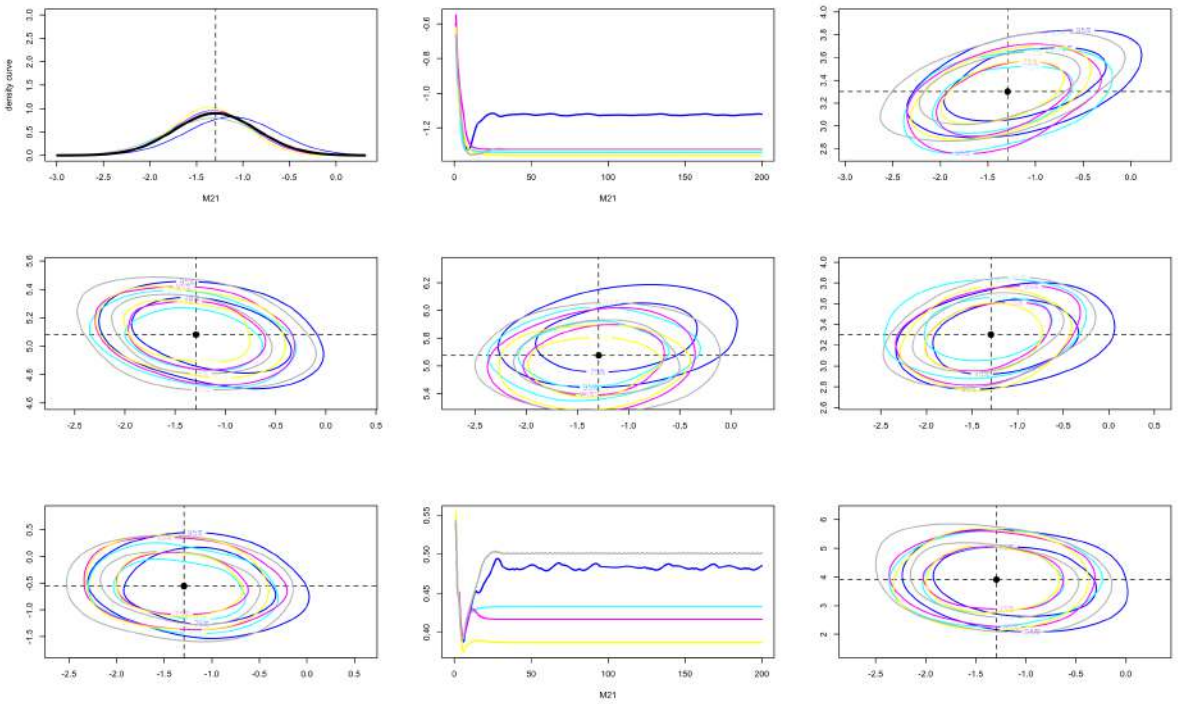

Figure 5.13: The EP method ABC Algorithm result for parameter $\phi^{[6]}$. The black curve and points present the mean of estimates from all 5 replicates. The coloured lines represent estimates from different replicates. The first plot in the top row is the marginal density curve of $\phi^{[6]}$ and the black curve represents the average estimates of parameter. The second plot in top row is the convergent performance for mean of parameter $\phi^{[6]}$. The remaining plots shows the 95% and 75% HPD curve between $\phi^{[6]}$ with $\phi^{[i]}$. If $i = 6$, then plot the convergent performance for the standard deviation of parameter $\phi^{[6]}$

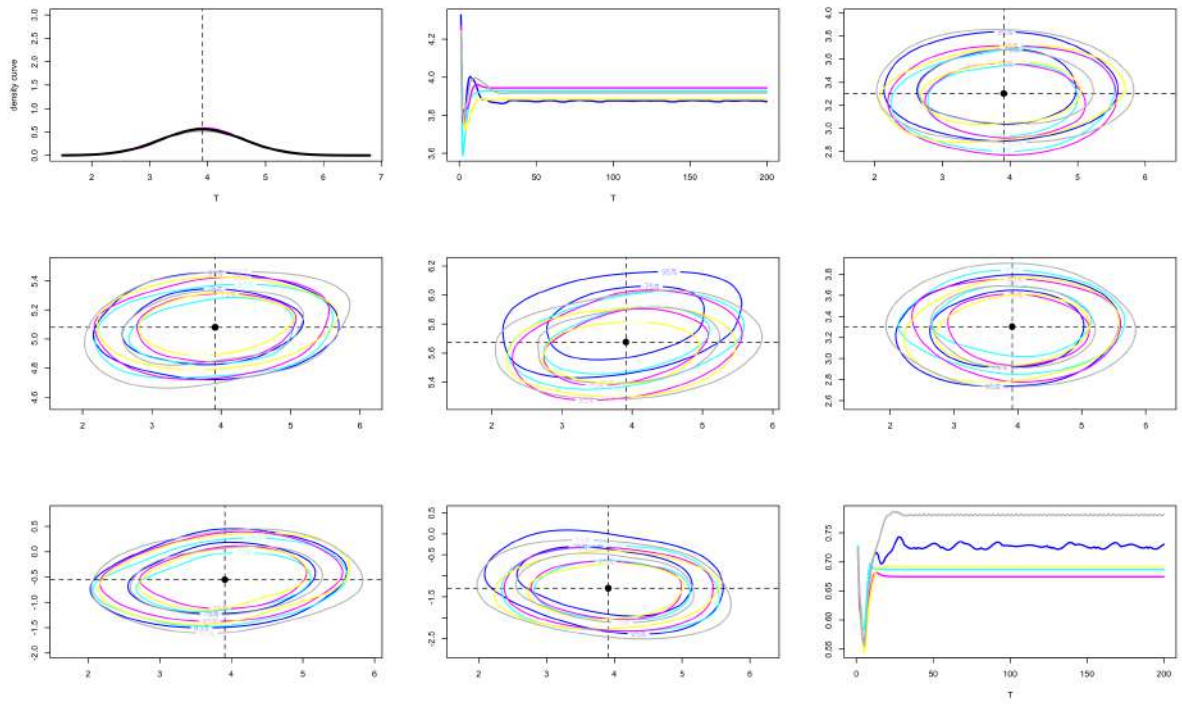

Figure 5.14: The EP method ABC Algorithm result for parameter $\phi^{[7]}$. The black curve and points present the mean of estimates from all 5 replicates. The coloured lines represent estimates from different replicates. The first plot in the top row is the marginal density curve of $\phi^{[7]}$ and the black curve represents the average estimates of parameter. The second plot in top row is the convergent performance for mean of parameter $\phi^{[7]}$. The remaining plots shows the 95% and 75% HPD curve between $\phi^{[7]}$ with $\phi^{[i]}$. If $i = 7$, then plot the convergent performance for the standard deviation of parameter $\phi^{[7]}$

CHAPTER 6

CONCLUSION

In this thesis, I mainly propose three Bayesian methods that can be employed to analysis time-series allele frequency data and contemporary whole-genome data. For each method, I present results from simulation studies to show the accuracy of inference and then apply it to analyze real data.

In Chapter 1, I introduce the basic concept of natural selection and the importance of understanding it. After that, I introduce the most important model used in my thesis, the Wright-Fisher model, and the most frequently used statistic techniques, Monte Carlo method. I list existed applications of them and present the key innovation to use them in population genetic problems. Besides that, I briefly discuss the time-series data and the utility of it. In this chapter, I try to make a connection between the population genetic problem with the Monte Carlo based statistical inference method.

In Chapter 2, I give a brief introduction of the Monte Carlo methods and mainly discuss MCMC techniques and ABC framework, which are highly related to my later chapters. I introduce Metropolis-Hastings firstly and focus on the pseudo-marginal Metropolis-Hastings algorithm that can use an unbiased approximate likelihood in the Metropolis-Hastings algorithm to solve intractable likelihood problems and illustrate the use of a pseudo-marginal Metropolis-Hastings algorithm with a Gaussian distribution example. Besides that, I also introduce existing ABC methods and illustrate how summary statistics affect the performance of ABC with a simulated Gaussian distribution example using Inverse-Gamma as a conjugate prior. Based on this simulated example, I also present the semi-automatic regression method [35] and the local linear regression method [15] which improve the performance of the ABC method. In this chapter, I use a variety of Gaussian distribution examples to illustrate the intuitive idea of different Monte Carlo based approaches. My aim is to use as simple examples as possible to illustrate the intuitive

idea. However, those examples seem to be irrelevance to the population genetics problem and lead this chapter statistical and theatrical.

In Chapter 3, I propose a particle marginal Metropolis-Hastings (PMMH) algorithm based on Andrieu et al. [3] to co-estimate the selection coefficient and allele age from an allele frequency trajectory. The intuitive idea is a two-step algorithm, we firstly co-estimate the selection coefficient and the initial population frequency using PMMH and using the estimates from the first step to infer allele age by solving the Kolmogorov Backward equation. From a simulation study, I show that the two-step method can achieve effective and accurate estimates for both the selection coefficient and allele age. I use the two-step method to re-analyze ancient DNA data with horse coat coloration from Ludwig et al. [71] and deliver a similar result with Ludwig et al. [71]. The two-step algorithm is simulation-based method, we need plenty of simulations involved in MCMC algorithm to make sure we have proper estimated distribution of the initial allele frequency, which is the link between the inference of the selection coefficient and the allele age. This process is very time-consuming but in fact, the initial allele frequency is not our parameter of genetics interest. It is redundancy in some degree and leaves motivation to construct a better statistical inference structure for such problem.

In Chapter 4, I present a two-locus Bayesian inference method based on PMMH and Wright-Fisher diffusion [52]. By taking local linkage and genetic recombination into account, the two-locus method can achieve more appropriate joint estimates of selection coefficients. I show a series of simulation studies to illustrate this method can properly infer selection given data with or without missing values. In addition, I compare the two-locus method with a single-locus method when local linkage and genetic recombination are needed to be considered, in that case, the two-locus method is more reliable than the single-locus method. I use this two-locus method to re-analyse the equine homologue of proto-oncogene c-kit (KIT) data published by Ludwig et al. [71]. In this real data, the two loci KIT13 and KIT16 are tightly linked with 4688 base pairs, which means taking local linkage and genetic recombination into account is necessary for inferring the selection coefficients of the mutant allele on the two loci. The result provides strong evidence that the mutant on KIT16 is to negatively selected based on the two-locus method, however, using a single-locus method, we can not detect evidence of negative selection. The Bayesian inference result of the KIT analysis is compatible with [128] allele frequency analysis. In the future, we aim to find statistical approaches that can handle data from more than two linked loci. The challenge is that as the number of loci increases, modelling the underlying population dynamics becomes increasingly difficult. For example, there are eight haplotypes to consider in the case of three loci each with two alleles. In practice, it will be necessary to find good approximations in order for the method to be computationally feasible. Recently there have been proposed some new methods that are worthy of investigating further, which can speed-up the inference process and improve the performance of estimates. For example, Scalable Monte Carlo [129], and the adaptive chain with early rejection method [107] are quite good choices to

combine with the method presented in Chapter 3 and 4.

In Chapter 5, I propose an Expectation Propagation method with an ABC algorithm based on a previous study of Mark Beaumont presented in the dissertation of William Perry [90], using the EP-ABC algorithm proposed by Barthelmé and Chopin [7]. The main point is to employ the ABC algorithm to make inferences based on decomposing genome data to form independent 'site' and using the EP method to combine these estimates to obtain the global estimates of genetic parameters of interest. By using this method, I uncover the demographic history of East African cichlids using whole-genome data [75]. The results are similar to the previous study presented in Malinsky et al. [75] and Perry and Beaumont [90]. In this chapter, the algorithm is tailored to dealing with population structure problems. Under such circumstances, I omit the selection process and regard all SNPs are neutral which is unrealistic. In the future, it is worthy to find some possible approaches to apply the machine learning idea to a more complex problem, for example, to jointly infer the selection process and the population structure. In addition, the EP method still suffers from no guarantee of convergence, and it may be worth exploring further stochastic natural gradient expectation (SNEP) [49] instead of the original EP structure for improved performance in parameter inference.

APPENDIX A

APPENDIX A

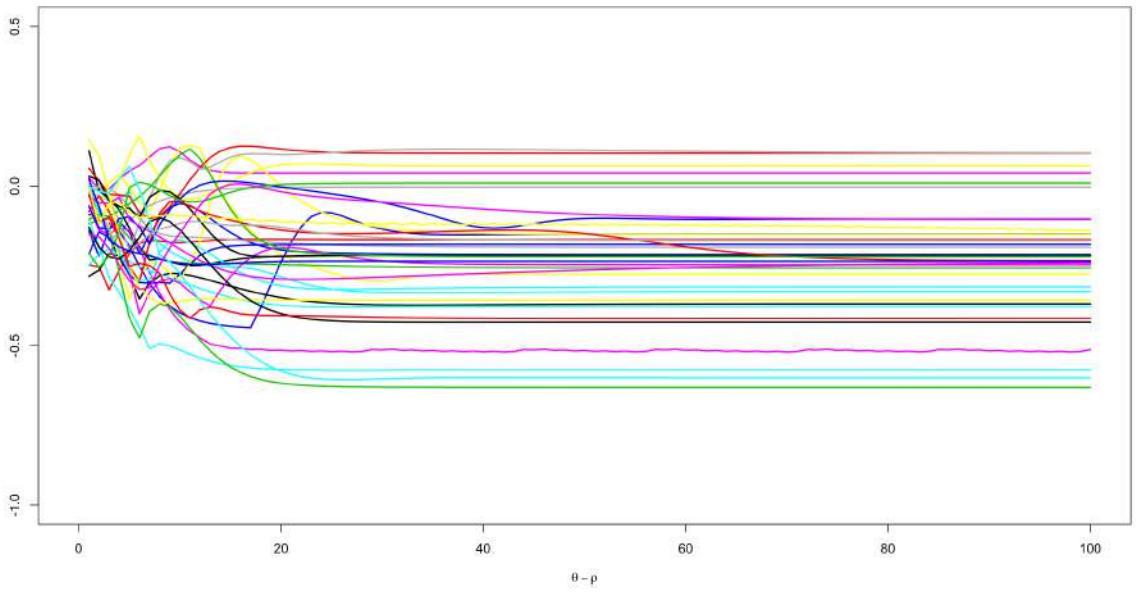

Figure A.1: The converge performance of element in covariance matrix.

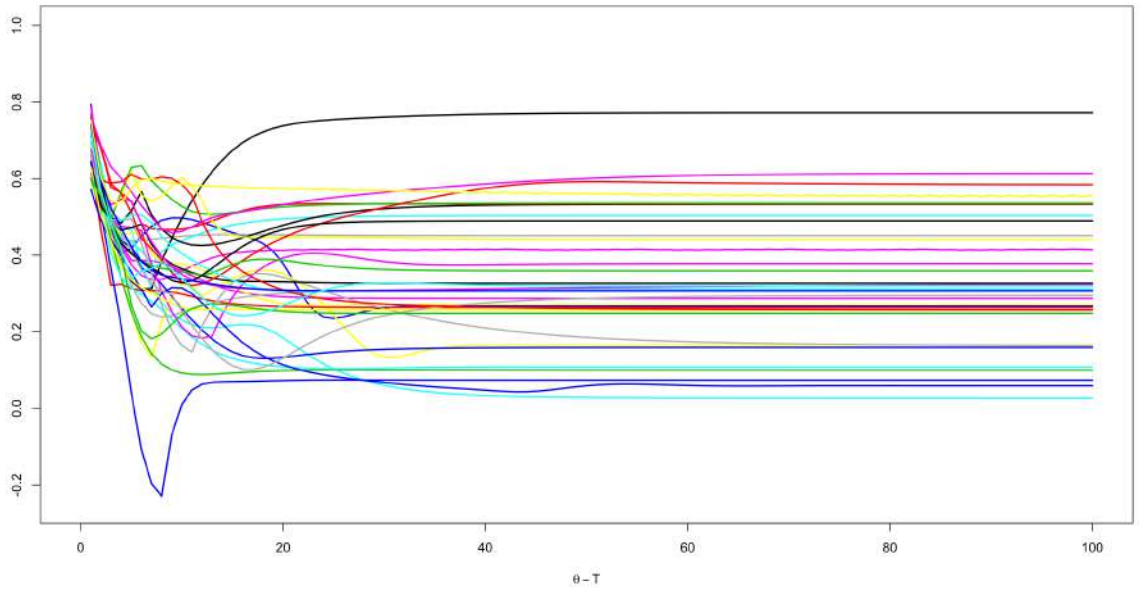

Figure A.2: The converge performance of element in covariance matrix.

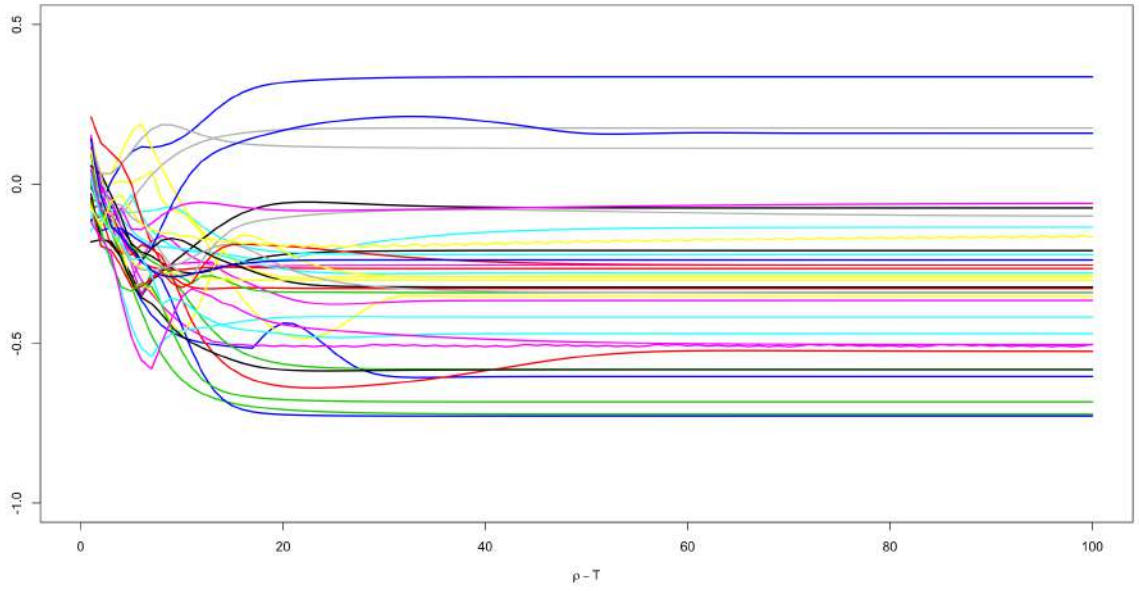

Figure A.3: The converge performance of element in covariance matrix.

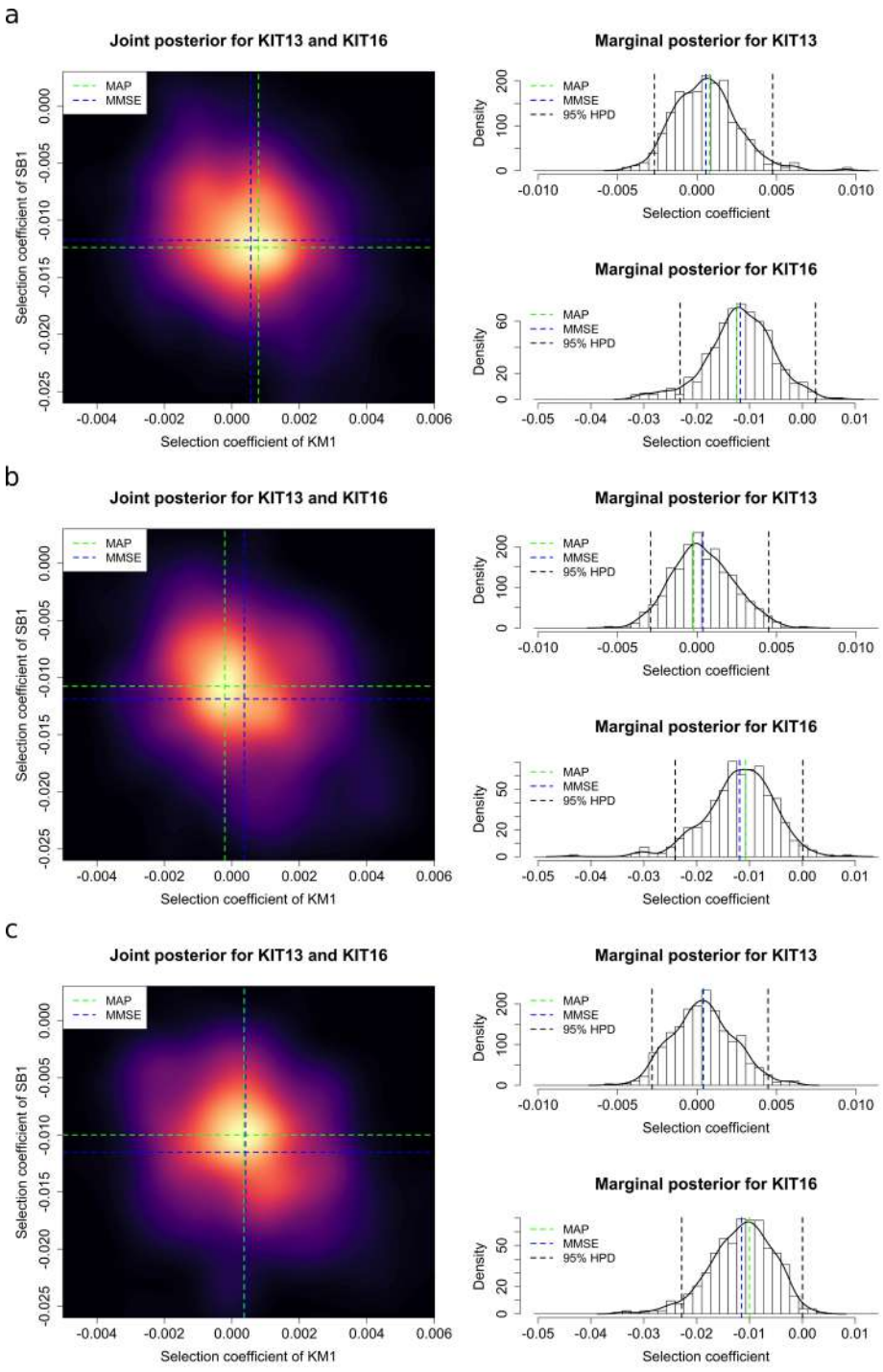

Figure A.4: Posterior probability distributions for *KIT13* and *KIT16* obtained with the population size of 16000 from the samples dated from 17146 years BP. (a) Posterior probability distributions with the average rate of recombination 5×10^{-9} crossovers/bp. (b) Posterior probability distributions with the average rate of recombination 1×10^{-8} crossovers/bp. (c) Posterior probability distributions with the average rate of recombination 5×10^{-8} crossovers/bp

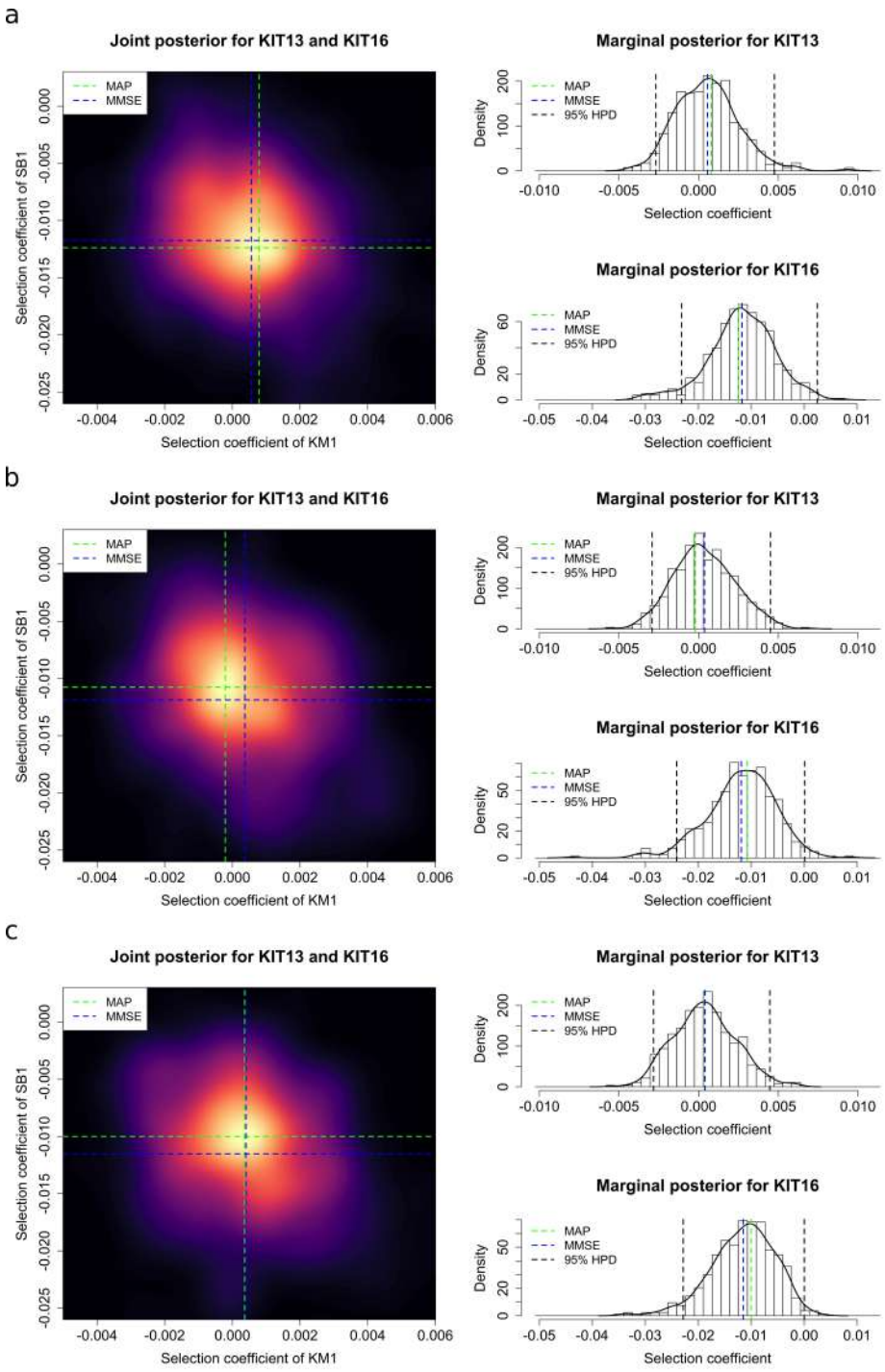

Figure A.5: Posterior probability distributions for *KIT13* and *KIT16* obtained with the population size of 16000 from the samples dated from 7029 years BP. (a) Posterior probability distributions with the average rate of recombination 5×10^{-9} crossovers/bp. (b) Posterior probability distributions with the average rate of recombination 1×10^{-8} crossovers/bp. (c) Posterior probability distributions with the average rate of recombination 5×10^{-8} crossovers/bp.

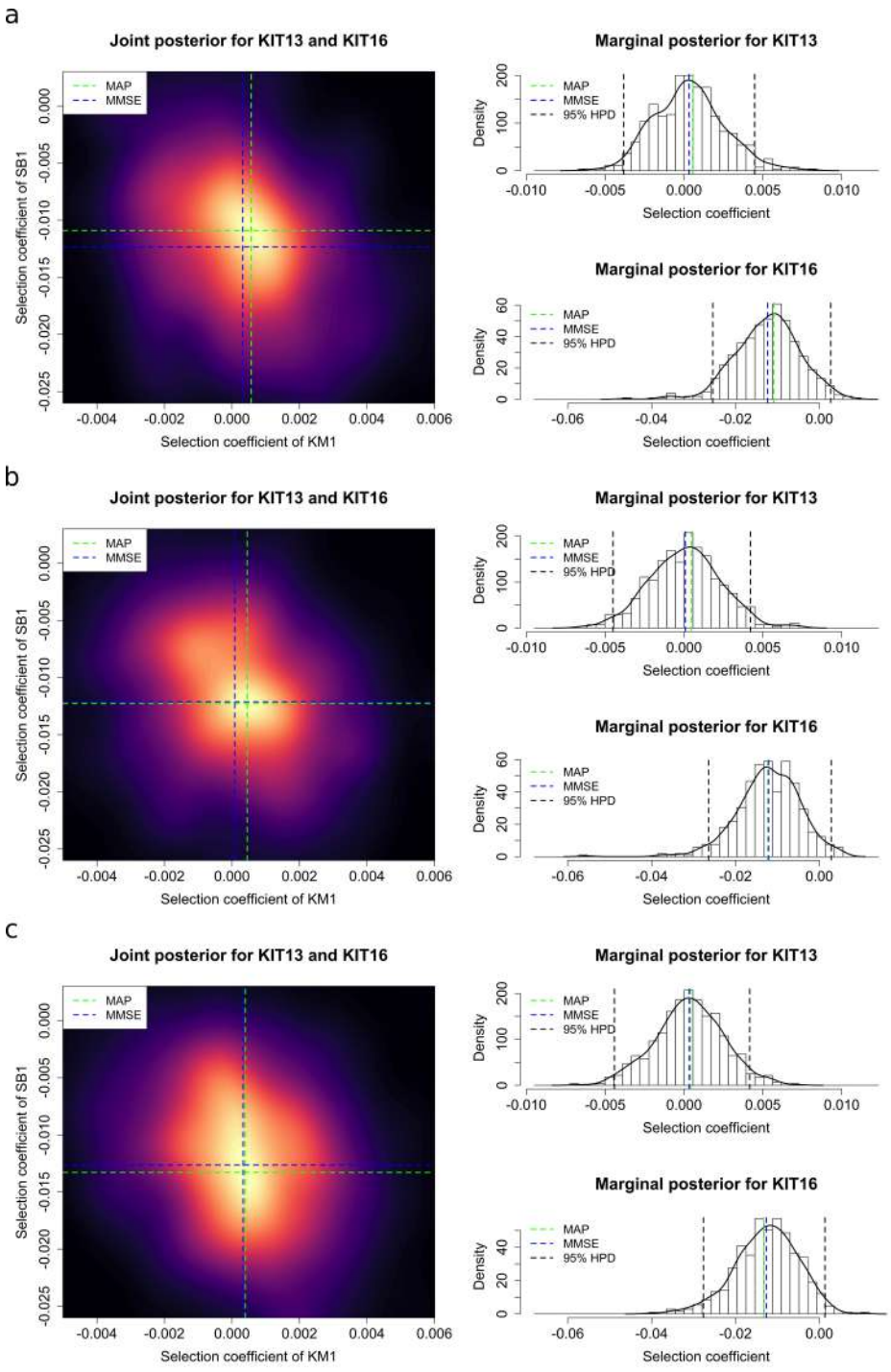

Figure A.6: Posterior probability distributions for *KIT13* and *KIT16* obtained with the population size of 8000 from the samples dated from 5472 years BP. (a) Posterior probability distributions with the average rate of recombination 5×10^{-9} crossovers/bp. (b) Posterior probability distributions with the average rate of recombination 1×10^{-8} crossovers/bp. (c) Posterior probability distributions with the average rate of recombination 5×10^{-8} crossovers/bp.

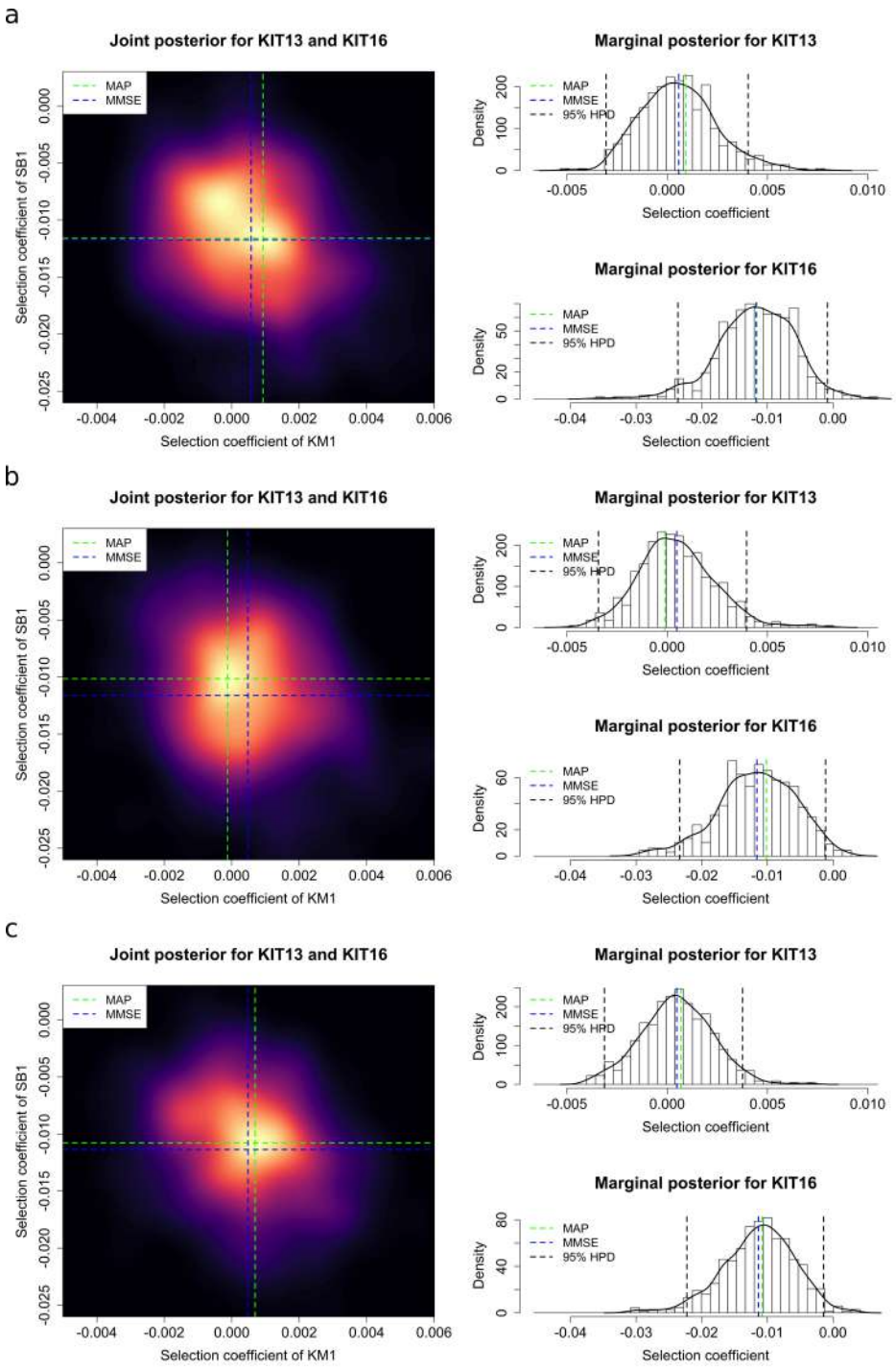

Figure A.7: Posterior probability distributions for *KIT13* and *KIT16* obtained with the population size of 32000 from the samples dated from 5472 years BP. (a) Posterior probability distributions with the average rate of recombination 5×10^{-9} crossovers/bp. (b) Posterior probability distributions with the average rate of recombination 1×10^{-8} crossovers/bp. (c) Posterior probability distributions with the average rate of recombination 5×10^{-8} crossovers/bp

BIBLIOGRAPHY

[1] Allentoft, M. E., Sikora, M., Sjögren, K.-G., Rasmussen, S., Rasmussen, M., Stenderup, J., Damgaard, P. B., Schroeder, H., Ahlström, T., Vinner, L., et al. (2015).
Population genomics of bronze age eurasia.
Nature, 522(7555):167.

[2] Anderson, E. C. (2001).
Monte Carlo methods for inference in population genetic models.
PhD thesis.

[3] Andrieu, C., Doucet, A., and Holenstein, R. (2010).
Particle Markov chain Monte Carlo methods.
Journal of the Royal Statistical Society: Series B (Statistical Methodology), 72(3):269–342.

[4] Andrieu, C. and Roberts, G. O. (2009).
The pseudo-marginal approach for efficient Monte Carlo computations.
The Annals of Statistics, 37(2):697–725.

[5] Andrieu, C., Roberts, G. O., et al. (2009).
The pseudo-marginal approach for efficient monte carlo computations.
The Annals of Statistics, 37(2):697–725.

[6] Andrieu, C. and Vihola, M. (2016).
Establishing some order amongst exact approximations of MCMCs.
The Annals of Applied Probability, 26(5):2661–2696.

[7] Barthelmé, S. and Chopin, N. (2014).
Expectation propagation for likelihood-free inference.
Journal of the American Statistical Association, 109(505):315–333.

[8] Barthelmé, S., Chopin, N., and Cottet, V. (2018).
Divide and conquer in abc: Expectation-propagation algorithms for likelihood-free inference.
Handbook of Approximate Bayesian Computation, pages 415–34.

[9] Bayram, M., Partal, T., and Buyukoz, G. O. (2018).
Numerical methods for simulation of stochastic differential equations.

Advances in Difference Equations, 2018(1):1–10.

[10] Beaumont, M. A. (2003a).
Estimation of population growth or decline in genetically monitored populations.
Genetics, 164(3):1139–1160.

[11] Beaumont, M. A. (2003b).
Estimation of population growth or decline in genetically monitored populations.
Genetics, 164(3):1139–1160.

[12] Beaumont, M. A. (2010).
Approximate bayesian computation in evolution and ecology.
Annual review of ecology, evolution, and systematics, 41:379–406.

[13] Beaumont, M. A. (2019).
Approximate bayesian computation.
Annual review of statistics and its application.

[14] Beaumont, M. A., Cornuet, J.-M., Marin, J.-M., and Robert, C. P. (2009).
Adaptive approximate bayesian computation.
Biometrika, 96(4):983–990.

[15] Beaumont, M. A., Zhang, W., and Balding, D. J. (2002).
Approximate bayesian computation in population genetics.
Genetics, 162(4):2025–2035.

[16] Beerli, P. and Felsenstein, J. (2001).
Maximum likelihood estimation of a migration matrix and effective population sizes in n subpopulations by using a coalescent approach.
Proceedings of the National Academy of Sciences, 98(8):4563–4568.

[17] Bollback, J. P., York, T. L., and Nielsen, R. (2008).
Estimation of $2N_e$s from temporal allele frequency data.
Genetics, 179(1):497–502.

[18] Brooks, S. A. and Bailey, E. (2005).
Exon skipping in the KIT gene causes a Sabino spotting pattern in horses.
Mammalian Genome, 16(11):893–902.

[19] Brooks, S. A., Lear, T. L., Adelson, D. L., and Bailey, E. (2007).
A chromosome inversion near the KIT gene and the Tobiano spotting pattern in horses.
Cytogenetic and Genome Research, 119(3-4):225–230.

[20] Burke, M. K., Dunham, J. P., Shahrestani, P., Thornton, K. R., Rose, M. R., and Long, A. D. (2010).
Genome-wide analysis of a long-term evolution experiment with drosophila.
Nature, 467(7315):587.

[21] Crow, J. F., Kimura, M., et al. (1970).
An introduction to population genetics theory.
An introduction to population genetics theory.

[22] Cui, T., Peeters, L., Pagendam, D., Pickett, T., Jin, H., Crosbie, R. S., Raiber, M., Rassam, D. W., and Gilfedder, M. (2018).
Emulator-enabled approximate bayesian computation (abc) and uncertainty analysis for computationally expensive groundwater models.
Journal of hydrology, 564:191–207.

[23] Cuthbertson, C., Etheridge, A., and Yu, F. (2012).
Fixation probability for competing selective sweeps.
Electronic Journal of Probability, 17(31):1–36.

[24] Darwin, C. (1859).
The Origin of Species; And, the Descent of Man.
Modern library.

[25] Dehaene, G. and Barthelmé, S. (2018).
Expectation propagation in the large data limit.
Journal of the Royal Statistical Society: Series B (Statistical Methodology), 80(1):199–217.

[26] Der Sarkissian, C., Ermini, L., Schubert, M., Yang, M. A., Librado, P., Fumagalli, M., Jónsson, H., Bar-Gal, G. K., Albrechtsen, A., Vieira, F. G., et al. (2015a).
Evolutionary genomics and conservation of the endangered przewalski's horse.
Current Biology, 25(19):2577–2583.

[27] Der Sarkissian, C., Ermini, L., Schubert, M., Yang, M. A., Librado, P., Fumagalli, M., Jónsson, H., Bar-Gal, G. K., Albrechtsen, A., Vieira, F. G., Petersen, B., Ginolhac, A., Seguin-Orlando, A., Magnussen, K., Fages, A., Gamba, C., Lorente-Galdos, B., Polani, S., Steiner, C., Neuditschko, M., Jagannathan, V., Feh, C., Greenblatt, C. L., Ludwig, A., Abramson, N. I., Zimmermann, W., Schafberg, R., Tikhonov, A., Sicheritz-Ponten, T., Willerslev, E., Marques-Bonet, T., Ryder, O. A., McCue, M., Rieder, S., Leeb, T., Slatkin, M., and Orlando, L. (2015b).
Evolutionary genomics and conservation of the endangered Przewalski's horse.
Current Biology, 25(19):2577–2583.

[28] Doucet, A., Pitt, M. K., Deligiannidis, G., and Kohn, R. (2015).
Efficient implementation of markov chain monte carlo when using an unbiased likelihood estimator.
Biometrika, 102(2):295–313.

[29] Dumont, B. L. and Payseur, B. A. (2008).
Evolution of the genomic rate of recombination in mammals.
Evolution, 62(2):276–294.

[30] Durrett, R. (2008a).
Probability Models for DNA Sequence Evolution.
Springer-Verlag, New York.

[31] Durrett, R. (2008b).
Probability models for DNA sequence evolution.
Springer Science & Business Media.

[32] Ewing, G. and Hermisson, J. (2010).
Msms: a coalescent simulation program including recombination, demographic structure and selection at a single locus.
Bioinformatics, 26(16):2064–2065.

[33] Fay, J. C. and Wu, C.-I. (2000).
Hitchhiking under positive darwinian selection.
Genetics, 155(3):1405–1413.

[34] Fearnhead, P. and Künsch, H. R. (2018).
Particle filters and data assimilation.
Annual Review of Statistics and Its Application, 5:421–449.

[35] Fearnhead, P. and Prangle, D. (2012).
Constructing summary statistics for approximate bayesian computation: semi-automatic approximate bayesian computation.
Journal of the Royal Statistical Society: Series B (Statistical Methodology), 74(3):419–474.

[36] Feder, A. F., Kryazhimskiy, S., and Plotkin, J. B. (2014).
Identifying signatures of selection in genetic time series.
Genetics, 196(2):509–522.

[37] Ferrer-Admetlla, A., Leuenberger, C., Jensen, J. D., and Wegmann, D. (2016).
An approximate Markov model for the Wright-Fisher diffusion and its application to time series data.
Genetics, 203(2):831–846.

[38] Fisher, R. A. (1922).
On the dominance ratio.
Proceedings of the Royal Society of Edinburgh, 42:321–341.

[39] Fisher, R. A. (1999).
The genetical theory of natural selection: a complete variorum edition.
Oxford University Press.

[40] Foll, M., Poh, Y.-P., Renzette, N., Ferrer-Admetlla, A., Bank, C., Shim, H., Malaspinas, A.-S., Ewing, G., Liu, P., Wegmann, D., Caffrey, D. R., Zeldovich, K. B., Bolon, D. N., Wang, J. P., Kowalik, T. F., Schiffer, C. A., Finberg, R. W., and Jensen, J. D. (2014).
Influenza virus drug resistance: a time-sampled population genetics perspective.
PLoS Genetics, 10(2):e1004185.

[41] Foll, M., Shim, H., and Jensen, J. D. (2015).
WFABC: a Wright-Fisher ABC-based approach for inferring effective population sizes and selection coefficients from time-sampled data.
Molecular Ecology Resources, 15(1):87–98.

[42] Gelman, A., Carlin, J. B., Stern, H. S., and Rubin, D. (2004).
Bayesian data analysis 2nd edn chapman & hall.
CRC, Boca Raton FL.

[43] Gelman, A., Vehtari, A., Jylänki, P., Sivula, T., Tran, D., Sahai, S., Blomstedt, P., Cunningham, J. P., Schiminovich, D., and Robert, C. (2017).
Expectation propagation as a way of life: A framework for bayesian inference on partitioned data.
arXiv preprint arXiv:1412.4869.

[44] Gikhman, I. I. and Skorokhod, A. V. (2007).
Stochastic differential equations.
In *The Theory of Stochastic Processes III*, pages 113–219. Springer.

[45] Gordon, N. J., Salmond, D. J., and Smith, A. F. M. (1993).
Novel approach to nonlinear/non-Gaussian Bayesian state estimation.
IEE Proceedings F (Radar and Signal Processing), 140(2):107–113.

[46] Gregory, T. R. (2009).
Understanding natural selection: essential concepts and common misconceptions.
Evolution: Education and outreach, 2(2):156.

[47] Gutenkunst, R. N., Hernandez, R. D., Williamson, S. H., and Bustamante, C. D. (2009).

Inferring the joint demographic history of multiple populations from multidimensional SNP frequency data.
PLoS Genetics, 5(10):e1000695.

[48] Hahn, C., Vakili, M., Walsh, K., Hearin, A. P., Hogg, D. W., and Campbell, D. (2017).
Approximate bayesian computation in large-scale structure: constraining the galaxy–halo connection.
Monthly Notices of the Royal Astronomical Society, 469(3):2791–2805.

[49] Hasenclever, L., Webb, S., Lienart, T., Vollmer, S., Lakshminarayanan, B., Blundell, C., and Teh, Y. W. (2017).
Distributed bayesian learning with stochastic natural gradient expectation propagation and the posterior server.
The Journal of Machine Learning Research, 18(1):3744–3780.

[50] He, Z., Beaumont, M. A., and Yu, F. (2017).
Effects of the ordering of natural selection and population regulation mechanisms on Wright-Fisher models.
G3: Genes, Genomes, Genetics, 7(7):2095–2106.

[51] He, Z., Beaumont, M. A., and Yu, F. (2019a).
A numerical solution of the Wright-Fisher stochastic differential equation with application to transition probability density approximation.
Manuscript submitted for publication.

[52] He, Z., Dai, X., Beaumont, M. A., and Yu, F. (2019b).
Maximum likelihood estimation of natural selection and allele age from time series data of allele frequencies.
Manuscript submitted for publication.

[53] Hey, J. and Nielsen, R. (2007).
Integration within the felsenstein equation for improved markov chain monte carlo methods in population genetics.
Proceedings of the National Academy of Sciences, 104(8):2785–2790.

[54] Holden, P. B., Edwards, N. R., Hensman, J., and Wilkinson, R. D. (2018).
Abc for climate: dealing with expensive simulators.
Handbook of Approximate Bayesian Computation, pages 569–95.

[55] Hudson, R. R. (2002).
Generating samples under a wright–fisher neutral model of genetic variation.
Bioinformatics, 18(2):337–338.

[56] Izenman, A. J. (1991).
Review papers: Recent developments in nonparametric density estimation.
Journal of the American Statistical Association, 86(413):205–224.

[57] Jabot, F. and Lohier, T. (2016).
Non-random correlation of species dynamics in tropical tree communities.
Oikos, 125(12):1733–1742.

[58] Jay, F., Boitard, S., and Austerlitz, F. (2019).
An abc method for whole-genome sequence data: inferring paleolithic and neolithic human expansions.
Molecular biology and evolution, 36(7):1565–1579.

[59] Joyce, P. and Marjoram, P. (2008).
Approximately sufficient statistics and bayesian computation.
Statistical applications in genetics and molecular biology, 7(1).

[60] Kandler, A. and Powell, A. (2018).
Generative inference for cultural evolution.
Philosophical Transactions of the Royal Society B: Biological Sciences, 373(1743):20170056.

[61] Kelleher, J., Etheridge, A. M., and McVean, G. (2016).
Efficient coalescent simulation and genealogical analysis for large sample sizes.
PLoS computational biology, 12(5):e1004842.

[62] Kimura, M. (1983).
The neutral theory of molecular evolution.
Cambridge University Press.

[63] Kimura, M. and Weiss, G. H. (1964).
The stepping stone model of population structure and the decrease of genetic correlation with distance.
Genetics, 49(4):561.

[64] Kloeden, P. E. and Platen, E. (1992).
Numerical Solution of Stochastic Differential Equations.
Springer-Verlag, Berlin.

[65] Kuhner, M. K., Yamato, J., and Felsenstein, J. (1998).
Maximum likelihood estimation of population growth rates based on the coalescent.
Genetics, 149(1):429–434.

[66] Lacerda, M. and Seoighe, C. (2014).

Population genetics inference for longitudinally-sampled mutants under strong selection.
Genetics, 198(3):1237–1250.

[67] Li, W. and Fearnhead, P. (2018).
On the asymptotic efficiency of approximate bayesian computation estimators.
Biometrika, 105(2):285–299.

[68] Li, Y., Hernández-Lobato, J. M., and Turner, R. E. (2015).
Stochastic expectation propagation.
In *Advances in neural information processing systems*, pages 2323–2331.

[69] Liepe, J., Kirk, P., Filippi, S., Toni, T., Barnes, C. P., and Stumpf, M. P. (2014).
A framework for parameter estimation and model selection from experimental data in systems biology using approximate bayesian computation.
Nature protocols, 9(2):439.

[70] Loog, L., Thomas, M. G., Barnett, R., Allen, R., Sykes, N., Paxinos, P. D., Lebrasseur, O., Dobney, K., Peters, J., Manica, A., et al. (2017).
Inferring allele frequency trajectories from ancient dna indicates that selection on a chicken gene coincided with changes in medieval husbandry practices.
Molecular biology and evolution, 34(8):1981–1990.

[71] Ludwig, A., Pruvost, M., Reissmann, M., Benecke, N., Brockmann, G. A., Castaños, P., Cieslak, M., Lippold, S., Llorente, L., Malaspinas, A.-S., et al. (2009a).
Coat color variation at the beginning of horse domestication.
Science, 324(5926):485–485.

[72] Ludwig, A., Pruvost, M., Reissmann, M., Benecke, N., Brockmann, G. A., Castaños, P., Cieslak, M., Lippold, S., Llorente, L., Malaspinas, A.-S., Slatkin, M., and Hofreiter, M. (2009b).
Coat color variation at the beginning of horse domestication.
Science, 324(5926):485–485.

[73] Malaspinas, A.-S. (2016).
Methods to characterize selective sweeps using time serial samples: an ancient DNA perspective.
Molecular Ecology, 25(1):24–41.

[74] Malaspinas, A.-S., Malaspinas, O., Evans, S. N., and Slatkin, M. (2012).
Estimating allele age and selection coefficient from time-serial data.
Genetics, 192(2):599–607.

[75] Malinsky, M., Challis, R. J., Tyers, A. M., Schiffels, S., Terai, Y., Ngatunga, B. P., Miska, E. A., Durbin, R., Genner, M. J., and Turner, G. F. (2015).
Genomic islands of speciation separate cichlid ecomorphs in an east african crater lake.
Science, 350(6267):1493–1498.

[76] Marjoram, P., Molitor, J., Plagnol, V., and Tavaré, S. (2003).
Markov chain monte carlo without likelihoods.
Proceedings of the National Academy of Sciences, 100(26):15324–15328.

[77] Mathieson, I., Lazaridis, I., Rohland, N., Mallick, S., Patterson, N., Roodenberg, S. A., Harney, E., Stewardson, K., Fernandes, D., Novak, M., Sirak, K., Gamba, C., Jones, E. R., Llamas, B., Dryomov, S., Pickrell, J., Arsuaga, J. L., de Castro, J. B., Carbonell, E., Gerritsen, F., Khokhlov, A., Kuznetsov, P., Lozano, M., Meller, H., Mochalov, O., Moiseyev, V., Guerra, M. A. R., Roodenberg, J., Vergés, J. M., Krause, J., Cooper, A., Alt, K. W., Brown, D., Anthony, D., Lalueza-Fox, C., Haak, W., Pinhasi, R., and Reich, D. (2015).
Genome-wide patterns of selection in 230 ancient Eurasians.
Nature, 528(7583):499–503.

[78] McKinley, T. J., Vernon, I., Andrianakis, I., McCreesh, N., Oakley, J. E., Nsubuga, R. N., Goldstein, M., White, R. G., et al. (2018).
Approximate bayesian computation and simulation-based inference for complex stochastic epidemic models.
Statistical science, 33(1):4–18.

[79] McVean, G. A. and Cardin, N. J. (2005).
Approximating the coalescent with recombination.
Philosophical Transactions of the Royal Society B: Biological Sciences, 360(1459):1387–1393.

[80] Metropolis, N. (1987).
The beginning of the.
Los Alamos Science, 15:125–30.

[81] Metropolis, N., Rosenbluth, A. W., Rosenbluth, M. N., Teller, A. H., and Teller, E. (1953).
Equation of state calculations by fast computing machines.
The journal of chemical physics, 21(6):1087–1092.

[82] Middleton, L., Deligiannidis, G., Doucet, A., and Jacob, P. E. (2018).
Unbiased markov chain monte carlo for intractable target distributions.
arXiv preprint arXiv:1807.08691.

[83] Mil'shtein, G. (1979).
A method of second-order accuracy integration of stochastic differential equations.
Theory of Probability & Its Applications, 23(2):396–401.

[84] Minka, T. P. (2001).
Expectation propagation for approximate bayesian inference.
In *Proceedings of the Seventeenth conference on Uncertainty in artificial intelligence*, pages 362–369. Morgan Kaufmann Publishers Inc.

[85] Morley, M., Molony, C. M., Weber, T. M., Devlin, J. L., Ewens, K. G., Spielman, R. S., and Cheung, V. G. (2004).
Genetic analysis of genome-wide variation in human gene expression.
Nature, 430(7001):743–747.

[86] Nei, M. and Li, W.-H. (1979).
Mathematical model for studying genetic variation in terms of restriction endonucleases.
Proceedings of the National Academy of Sciences, 76(10):5269–5273.

[87] Nunes, M. A. and Balding, D. J. (2010).
On optimal selection of summary statistics for approximate bayesian computation.
Statistical applications in genetics and molecular biology, 9(1).

[88] Orlando, L., Ginolhac, A., Zhang, G., Froese, D., Albrechtsen, A., Stiller, M., Schubert, M., Cappellini, E., Petersen, B., Moltke, I., et al. (2013).
Recalibrating equus evolution using the genome sequence of an early middle pleistocene horse.
Nature, 499(7456):74.

[89] Outram, A. K., Stear, N. A., Bendrey, R., Olsen, S., Kasparov, A., Zaibert, V., Thorpe, N., and Evershed, R. P. (2009).
The earliest horse harnessing and milking.
Science, 323(5919):1332–1335.

[90] Perry, W. and Beaumont, M. (2016).
Insight into the demographic history of east african cichilids using bayesian computation and whole-genome data.
Fourth year project dissertation for Biological Sciences of University of Bristol, 1(1):1–40.

[91] Pritchard, J. K., Seielstad, M. T., Perez-Lezaun, A., and Feldman, M. W. (1999).
Population growth of human y chromosomes: a study of y chromosome microsatellites.
Molecular biology and evolution, 16(12):1791–1798.

[92] Pritchard, J. K., Stephens, M., and Donnelly, P. (2000).
Inference of population structure using multilocus genotype data.
Genetics, 155(2):945–959.

[93] Pruvost, M., Bellone, R., Benecke, N., Sandoval-Castellanos, E., Cieslak, M., Kuznetsova, T., Morales-Muñiz, A., O'Connor, T., Reissmann, M., Hofreiter, M., and Ludwig, A. (2011).

Genotypes of predomestic horses match phenotypes painted in Paleolithic works of cave art. *Proceedings of the National Academy of Sciences*, 108(46):18626–18630.

[94] Raynal, L., Marin, J.-M., Pudlo, P., Ribatet, M., Robert, C. P., and Estoup, A. (2018). Abc random forests for bayesian parameter inference. *Bioinformatics*, 35(10):1720–1728.

[95] Robert, C. and Casella, G. (2013). *Monte Carlo statistical methods*. Springer Science & Business Media.

[96] Rubin, D. B. et al. (1984). Bayesianly justifiable and relevant frequency calculations for the applied statistician. *The Annals of Statistics*, 12(4):1151–1172.

[97] Rydén, T. (1994). Consistent and asymptotically normal parameter estimates for hidden markov models. *The Annals of Statistics*, pages 1884–1895.

[98] Schmon, S. M., Deligiannidis, G., Doucet, A., and Pitt, M. K. (2018). Large sample asymptotics of the pseudo-marginal method. *arXiv preprint arXiv:1806.10060*.

[99] Schraiber, J. G., Evans, S. N., and Slatkin, M. (2016). Bayesian inference of natural selection from allele frequency time series. *Genetics*, 203(1):493–511.

[100] Scott, S. L., Blocker, A. W., Bonassi, F. V., Chipman, H. A., George, E. I., and McCulloch, R. E. (2016). Bayes and big data: The consensus monte carlo algorithm. *International Journal of Management Science and Engineering Management*, 11(2):78–88.

[101] Seeger, M. and Nickisch, H. (2011). Fast convergent algorithms for expectation propagation approximate bayesian inference. In *Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics*, pages 652–660.

[102] Sheehan, S. and Song, Y. S. (2016). Deep learning for population genetic inference. *PLoS computational biology*, 12(3):e1004845.

[103] Shim, H., Laurent, S., Matuszewski, S., Foll, M., and Jensen, J. D. (2016). Detecting and quantifying changing selection intensities from time-sampled polymorphism data.

G3: Genes, Genomes, Genetics, 6(4):893–904.

[104] Sisson, S. A., Fan, Y., and Tanaka, M. M. (2007).
Sequential monte carlo without likelihoods.
Proceedings of the National Academy of Sciences, 104(6):1760–1765.

[105] Sjödin, P., E. Sjöstrand, A., Jakobsson, M., and Blum, M. G. (2012).
Resequencing data provide no evidence for a human bottleneck in africa during the penultimate glacial period.
Molecular biology and evolution, 29(7):1851–1860.

[106] Slatkin, M. and Rannala, B. (2000).
Estimating allele age.
Annual Review of Genomics and Human Genetics, 1(1):225–249.

[107] Solonen, A., Ollinaho, P., Laine, M., Haario, H., Tamminen, J., Järvinen, H., et al. (2012).
Efficient mcmc for climate model parameter estimation: Parallel adaptive chains and early rejection.
Bayesian Analysis, 7(3):715–736.

[108] Song, Y. S. and Steinrücken, M. (2012).
A simple method for finding explicit analytic transition densities of diffusion processes with general diploid selection.
Genetics, 190(3):1117–1129.

[109] Staab, P. R., Zhu, S., Metzler, D., and Lunter, G. (2015).
scrm: Efficiently simulating long sequences using the approximated coalescent with recombination.
Bioinformatics, 31(10):1680–1682.

[110] Steinrücken, M., Bhaskar, A., and Song, Y. S. (2014).
A novel spectral method for inferring general diploid selection from time series genetic data.
The Annals of Applied Statistics, 8(4):2203–2222.

[111] Stigler, S. M. (1973).
Studies in the history of probability and statistics. xxxii: Laplace, fisher, and the discovery of the concept of sufficiency.
Biometrika, 60(3):439–445.

[112] Tajima, F. (1989).
Statistical method for testing the neutral mutation hypothesis by dna polymorphism.
Genetics, 123(3):585–595.

[113] Tataru, P., Simonsen, M., Bataillon, T., and Hobolth, A. (2017).
Statistical inference in the Wright-Fisher model using allele frequency data.
Systematic Biology, 66(1):e30–e46.

[114] Teh, Y. W., Hasenclever, L., Lienart, T., Vollmer, S., Webb, S., Lakshminarayanan, B., and Blundell, C. (2015).
Distributed bayesian learning with stochastic natural-gradient expectation propagation and the posterior server.
arXiv preprint arXiv:1512.09327.

[115] Terhorst, J., Schlötterer, C., and Song, Y. S. (2015).
Multi-locus analysis of genomic time series data from experimental evolution.
PLoS Genetics, 11(4):e1005069–e1005069.

[116] Turner, B. M., Dennis, S., and Van Zandt, T. (2013).
Likelihood-free bayesian analysis of memory models.
Psychological review, 120(3):667.

[117] van der Vaart, E., Beaumont, M. A., Johnston, A. S., and Sibly, R. M. (2015).
Calibration and evaluation of individual-based models using approximate bayesian computation.
Ecological Modelling, 312:182–190.

[118] VanLiere, J. M. and Rosenberg, N. A. (2008).
Mathematical properties of the r2 measure of linkage disequilibrium.
Theoretical population biology, 74(1):130–137.

[119] Vehtari, A., Gelman, A., Sivula, T., Jylanki, P., Tran, D., Sahai, S., Blomstedt, P., Cunningham, J. P., Schiminovich, D., Robert, C. P., et al. (2020).
Expectation propagation as a way of life.

[120] Watterson, G. (1975).
On the number of segregating sites in genetical models without recombination.
Theoretical population biology, 7(2):256–276.

[121] Weiss, G. and von Haeseler, A. (1998).
Inference of population history using a likelihood approach.
Genetics, 149(3):1539–1546.

[122] Wichman, H. A., Millstein, J., and Bull, J. J. (2005).
Adaptive molecular evolution for 13,000 phage generations: a possible arms race.
Genetics, 170(1):19–31.

[123] Williamson, E. G. and Slatkin, M. (1999).
Using maximum likelihood to estimate population size from temporal changes in allele frequencies.
Genetics, 152(2):755–761.

[124] Wilson, G. A. and Rannala, B. (2003).
Bayesian inference of recent migration rates using multilocus genotypes.
Genetics, 163(3):1177–1191.

[125] Wilson, I. J. and Balding, D. J. (1998).
Genealogical inference from microsatellite data.
Genetics, 150(1):499–510.

[126] Wright, S. (1931).
Evolution in Mendelian populations.
Genetics, 16(2):97–159.

[127] Wright, S. (1984).
Evolution and the genetics of populations: genetics and biometric foundations vol. 2 (the theory of gene frequencies). u.

[128] Wutke, S., Benecke, N., Sandoval-Castellanos, E., Döhle, H.-J., Friederich, S., Gonzalez, J., Hallsson, J. H., Hofreiter, M., Lõugas, L., Magnell, O., Morales-Muniz, A., Orlando, L., Pálsdóttir, A. H., Reissmann, M., Ruttkay, M., Trinks, A., and Ludwig, A. (2016).
Spotted phenotypes in horses lost attractiveness in the Middle Ages.
Scientific Reports, 6:38548.

[129] Yıldırım, S., Andrieu, C., and Doucet, A. (2018).
Scalable monte carlo inference for state-space models.
arXiv preprint arXiv:1809.02527.

[130] Yu, F. and Etheridge, A. (2010).
The fixation probability of two competing beneficial mutations.
Theoretical Population Biology, 78(1):36–45.